

ALPHAMELTS SOFTWARE MANUAL.

A text driven interface for MELTS, pMELTS & pHMELTS.

Paula Antoshechkina (psmith@gps.caltech.edu) & Paul Asimow (asimow@gps.caltech.edu).

August 27, 2018 (Version 1.9).

ABSTRACT

The alphaMELTS software package and included Perl scripts provide a simple text-based interface to subroutine versions of the algorithms MELTS, pMELTS, and pHMELTS. Calculations may be performed interactively with simple double-clicking and drag-and-drop operations. Alternatively, a batch mode allows almost complete automation of the calculation process when suitable input files are written. alphaMELTS is available for Linux, Mac OS X, and Windows and output is easily imported into commercial software for plotting and analysis. A copy of the original technical brief (Smith & Asimow, 2005) from G-cubed is included in the software package. This documentation expands upon and updates the general information given there and then lists the program options and file formats in detail.

Introduction	5
Description	5
<i>Thermodynamic model</i>	5
<i>Alternative garnet, spinel, amphibole and biotite models</i>	6
<i>Bulk composition</i>	6
<i>Thermodynamic path</i>	7
<i>Starting conditions</i>	7
<i>Trace elements and water partitioning</i>	8
<i>Open versus closed system behavior</i>	8
<i>Special calculations</i>	9
<i>Output</i>	10
How to cite alphaMELTS	10
Setting up alphaMELTS	11
<i>Before you start</i>	11
<i>Installation</i>	12
<i>Interactive operation</i>	15
Using alphaMELTS	17
<i>Synopsis</i>	17
<i>Input file formats</i>	18
MELTS file.....	18
Enrichment file.....	21
Table file.....	21
Trace Data file.....	22
Settings file.....	24
Batch file.....	25
<i>Output files</i>	25
Main output files.....	25
Integrated output file.....	27
Thermodynamic phase output file.....	28
<i>Binary restart files</i>	29
Restart file use: isenthalpic assimilation.....	30
Restart file use: phase metastability.....	30
<i>Command line switches</i>	30
-h.....	30
-f <i>settings_file</i>	31
-b <i>batch_file</i>	31
-a.....	31
-m <i>melts_file</i>	31
-t <i>table_file</i>	32
-o <i>output_file</i>	32
-p <i>output_path</i>	32
-c <i>column_file</i>	32
-l <i>log_file</i>	33
<i>Runtime alphamelts menu options</i>	33
1. Read MELTS file to set composition of system.....	33
2. Twiddle starting or continuation parameters.....	33
3. Single (batch) calculation.....	33
4. Execute (follow path, mineral isograd or melt contour).....	34
5. Set fO2 buffer.....	34

6. Set H2O (ppm) or aH2O.....	35
7. Impose initial entropy, enthalpy or volume.....	35
8. Adjust solid phase setting(s)	35
9. Turn liquid on / off.....	35
10. Turn phase diagram mode on / off.....	35
11. Update state using restart file	36
12. Update composition using MELTS file	37
13. Write out restart file	37
14. Write out MELTS file	37
15. Write thermodynamic output for all phases.....	38
16. Calculate integrated melt and output file(s)	38
17. Fit parental melt composition (amoeba).....	39
18. Cumulate Invertor (not yet implemented).....	39
-1. Turn off (or on) menu display for options 1-18	40
0. Quit.....	40
<i>Note about initial P, T, etc.....</i>	<i>40</i>
Environment Variables.....	40
<i>Solution models.....</i>	<i>40</i>
ALPHAMELTS_VERSION	40
ALPHAMELTS_OLD_GARNET.....	40
ALPHAMELTS_OLD_SPINEL.....	40
ALPHAMELTS_OLD_BIOTITE.....	41
ALPHAMELTS_2_AMPH.....	41
ALPHAMELTS_NO_CHLORITE.....	41
<i>P-T path and fO2 control.....</i>	<i>41</i>
ALPHAMELTS_MODE	41
ALPHAMELTS_PTPATH_FILE	42
ALPHAMELTS_DELTAP	42
ALPHAMELTS_DELTAT	42
ALPHAMELTS_MAXP.....	42
ALPHAMELTS_MINP.....	42
ALPHAMELTS_MAXT.....	42
ALPHAMELTS_MINT.....	43
ALPHAMELTS_ALTERNATIVE_FO2	43
ALPHAMELTS_LIQUID_FO2.....	43
ALPHAMELTS_IMPOSE_FO2	43
ALPHAMELTS_FO2_PRESSURE_TERM	43
<i>Open vs. closed system.....</i>	<i>44</i>
ALPHAMELTS_CONTINUOUS_MELTING.....	44
ALPHAMELTS_MINF.....	44
ALPHAMELTS_MINPHI.....	44
ALPHAMELTS_CONTINUOUS_RATIO	44
ALPHAMELTS_CONTINUOUS_VOLUME	45
ALPHAMELTS_FRACTIONATE_SOLIDS	45
ALPHAMELTS_MASSIN.....	45
ALPHAMELTS_FRACTIONATE_WATER	45
ALPHAMELTS_MINW	45
ALPHAMELTS_FRACTIONATE_TARGET	46
ALPHAMELTS_MGO_TARGET.....	46
ALPHAMELTS_MGNUMBER_TARGET	47

ALPHAMEELTS_ASSIMILATE	47
ALPHAMEELTS_FLUX_MELTING	48
ALPHAMEELTS_DRY_ITER_PATIENCE.....	48
<i>pHMELTS and Trace Elements.....</i>	<i>49</i>
ALPHAMEELTS_DO_TRACE.....	49
ALPHAMEELTS_DO_TRACE_H2O.....	49
ALPHAMEELTS_HK_OL_TRACE_H2O	49
ALPHAMEELTS_HK_PXGT_TRACE_H2O.....	49
ALPHAMEELTS_2X_OPX_TRACE_H2O	49
ALPHAMEELTS_TRACE_DEFAULT_DPTX.....	50
ALPHAMEELTS_TRACE_NORMALIZATION	50
ALPHAMEELTS_TRACE_INPUT_FILE	50
ALPHAMEELTS_TRACE_USELIQFEMG	50
<i>Input options.....</i>	<i>51</i>
ALPHAMEELTS_ADIABAT_BIN_FILE.....	51
<i>Output options.....</i>	<i>51</i>
ALPHAMEELTS_CELSIUS_OUTPUT.....	51
ALPHAMEELTS_SAVE_ALL.....	51
ALPHAMEELTS_SKIP_FAILURE.....	51
ALPHAMEELTS_FAILED_ITER_PATIENCE.....	52
ALPHAMEELTS_INTEGRATE_FILE.....	52
ALPHAMEELTS_LATENT_HEAT.....	52
ALPHAMEELTS_QUICK_OUTPUT.....	52
<i>Others.....</i>	<i>53</i>
ALPHAMEELTS_MULTIPLE_LIQUIDS.....	53
ALPHAMEELTS_FRACTIONATE_SECOND_LIQUID	53
ALPHAMEELTS_FOCUS	53
ALPHAMEELTS_FOCUS_FACTOR.....	53
ALPHAMEELTS_INTEGRATE_PHI.....	53
The column_pick.command script.....	54
<i>Synopsis.....</i>	<i>54</i>
<i>Column list file.....</i>	<i>54</i>
What to do when (pH)MELTS misbehaves.....	56
Bugs and fixes.....	57
Disclaimer	58
Bibliography	59

Introduction

The alphaMELTS software package contains a simple text-menu driver for subroutine versions of the algorithms MELTS, pMELTS and pHMELTS [Asimow, *et al.*, 2004; Ghiorso, *et al.*, 2002; Ghiorso and Sack, 1995]. It may be used to calculate equilibrium assemblages along a thermodynamic path set by the user and can simultaneously calculate trace element distributions. The MELTS family of algorithms is suitable for multi-component systems, which may be anhydrous, hydrous or water undersaturated, with the options of buffering fO_2 and / or aH₂O. A wide variety of calculations can be performed either sub-solidus or with liquid(s) present; melting and crystallization may be batch, fractional or continuous. The available features for this command line based version are outlined in the following section; for more information see *Smith and Asimow* [2005]. Following the general description is a detailed list of the input and output files, options and switches and how to use them. For background on the MELTS, pMELTS and Rhyolite-MELTS software (GUI versions), please visit the <http://melts.ofm-research.org/> site. We also suggest that you browse the MELTS software users' forum at <https://magmasource.caltech.edu/forum/> for specific queries and examples.

alphaMELTS 1.9 can be downloaded from <https://magmasource.caltech.edu/alphamelts/> and is suitable for Linux, MacOS and Windows (64-bit processors). Virtual machine versions (VM alphaMELTS) are also available in 64-bit and 32-bit. This requires that VirtualBox program or similar is installed on the host machine, but if the Rhyolite-MELTS GUI, as well as alphaMELTS 1.9, 1.8.1, and 2.0 (beta) set up and ready to go. Native 32-bit executables are also available in alphaMELTS 1.8.1, which can be downloaded from the alphaMELTS archive.

In alphaMELTS 1 series, many aspects of program execution are controlled by environment variables. The Perl file `run_alphamelts.command` may be used to invoke the `alphamelts` executable with some command line options and automatically tidies up the output files. Benefits include a batch mode, which allows almost complete automation of the calculation process when suitable input files are written. alphaMELTS 2 (beta) is a complete rewrite of the underlying code, including the rhyolite-MELTS and mixed H₂O-CO₂ fluid models [Gualda, *et al.*, 2012; Ghiorso and Gualda, 2015]. A helper script, `run-alphamelts.command`, may still be used but environment variables are mostly replaced with text input. alphaMELTS 1.9 includes some backported features from version 2, to ease the transition as the latter becomes a more fully-developed package.

Description

Thermodynamic model

The underlying thermodynamic model behind the alphaMELTS front end is either MELTS [Ghiorso and Sack, 1995] or pMELTS [Ghiorso, *et al.*, 2002]. The former is calibrated on a wide variety of bulk compositions and is suitable at pressures not greater than 3 GPa, whereas the latter is intended for peridotite compositions only between 1 and 4 GPa. An environment variable may be used to specify the choice of thermodynamic model and default maximal pressure ranges will be adjusted to reflect this. We do not recommend using either model outside its intended composition or pressure range but note that the limits refer to the calibrated *liquid* model. The solid models used in MELTS and pMELTS are the same and so if, and only if, the liquid phase is suppressed then calculations between 1 bar and 4GPa are valid; this relaxation of the pressure limits allow a specific potential temperature to be imposed with a single pMELTS run (e.g. see the '[Exercises For MELTS](#)' tutorial, or more up to date workshop examples at <https://magmasource.caltech.edu/alphamelts/archive.php>).

In pHMELTS, the effects of storage in water in nominally anhydrous minerals are approximately modeled [Asimow, *et al.*, 2004; Smith and Asimow, 2005]. alphaMELTS can run MELTS or pMELTS in their conventional forms or the modified pHMELTS versions. For pHMELTS, the user should run in pMELTS mode for high-pressure melting calculations and MELTS mode for low-pressure crystallization (although during crystallization water behaves sufficiently incompatibly that the pHMELTS enhancement to MELTS is probably unnecessary). pHMELTS is intended to be used in isentropic or isobaric modes; it should work for other constraints such as isenthalpic and isochoric but this has not been tested. In this document, MELTS, pMELTS and / or pHMELTS are sometimes collectively called (pH)MELTS.

Alternative garnet, spinel, amphibole and biotite models

When *Adiabat_1ph* was released the original garnet model was retained as the default, despite containing a mistake, as it had been used in calibration of the MELTS and pMELTS liquid models [Smith and Asimow, 2005]. The corrected model was then available as an option. As all current GUI versions of MELTS and pMELTS have since been updated to use the corrected model this is the default in alphaMELTS. An option exists to use the older, incorrect, model; for more information on this and other environment variables in this section see '[Solution models](#)'.

There are also options to choose between slightly different spinel and amphibole models. The spinel in the GUI versions of MELTS, also the default for alphaMELTS, includes volume of mixing terms on the magnetite-ulvospinel join (Ghiorso and Sack [1991]; see Hamechar, *et al.* [2011] for discussion). Following the treatment in the standalone version of the MELTS software, alphaMELTS has a single cummingtonite-grunerite-tremolite solution, which may have a monoclinic and orthorhombic structure. For backwards compatibility (*Adiabat_1ph* prior to version 3.0) and consistency with some other MELTS GUIs, there are options to have separate clin amphibole and orthoamphibole phases and / or use the ideal spinel volume model.

The default biotite model included in alphaMELTS differs from that in the GUI versions of MELTS and includes data for annite and an ideal mixing model taken from McMullin, *et al.* [1991]. This revision generates reasonable Fe abundances in biotite, with the right temperature dependence. The biotite model from the GUI versions of MELTS is also available as an alternative. Although a small error in the coding has been corrected, small inconsistencies in the annite standard state properties [Berman, 1990] may still lead to dominance of the phlogopite component if this model is adopted. On the other hand, ideal mixing in biotite is almost certainly an oversimplification [Sack and Ghiorso, 1989] and the default model could have undesired effects on the Fe-Mg exchange between biotite and other solid phases. Fortunately, no biotite-liquid equilibria were included in the MELTS and pMELTS calibrations.

A full list of available solid solution phases and their end-members, in the order in which they are usually output, is available on the forum. Some of the end-member thermodynamic data differ slightly between MELTS and pMELTS, as described in Ghiorso, *et al.* [2002]. Note that alphaMELTS always uses pMELTS data for solid phases except with feldspar, for which it takes MELTS values. For any given software package or web applet, the solid solution model is exactly the same in the MELTS and pMELTS models (and therefore also in pHMELTS, which is based on pMELTS); this holds for all phases, not just garnet, biotite, spinel and amphibole.

Bulk composition

The major and trace element bulk composition is first read from a MELTS-style input file. The correct format for the file (referred to here as the *melts_file*) is specified below and the user may also

refer to example input files provided for a variety of commonly used bulk compositions. The `run_alphamelts.command` script has a `table_file` function for generating multiple `melts_files` easily. In addition, a second bulk composition file may be provided with a similar format to the `melts_file` (called the `enrichment_file`) and read into alphaMELTS via the menu (see option 12). In this case, a new source composition is calculated by mixing the `melts_file` composition and the `enrichment_file` composition in user-specified proportions. This option could be used to model, for example, melting of a (fully homogenized) mixture of peridotite and pyroxenite components. Isotopic ratios and water content of the new source can also be calculated in this way.

An extension of the available options is that a second bulk composition, using the same `enrichment_file` format, may be added at each stage of the calculation in order to simulate flux melting of a region or assimilation of wall rocks during crystallization. In the latter case, the temperature of the assimilant will be ignored unless the mode is isenthalpic / isentropic (likewise pressure for isochoric calculations). These routines were thoroughly overhauled for Adiabats_1ph version 3.0 to simplify the procedures and so that, for example, variable enrichment compositions may be used. As of alphaMELTS 1.3, it is also possible to specify individual mineral compositions and proportions within an assimilated lithology.

Thermodynamic path

The simplest form of calculation is along an adiabat (the default), an isotherm, an isobar or a geotherm. During normal execution, equilibria are calculated at equally spaced points in pressure (for isothermal, isentropic) or temperature (for isobaric) or both (for geothermal). For the isentropic mode, the user can either assign the starting total extensive entropy manually or give a prescribed temperature in which case the entropy of the resulting equilibrium becomes the reference entropy.

More unusual constraints include constant volume (isochoric), though this approximates quite well the conditions experienced by melt inclusions. We have also enabled the isenthalpic mode [Thompson, *et al.*, 2007], so that heat-balanced assimilation in an evolving magma chamber may be modeled in a similar way to the (GUI) MELTS calculations described by Reiners, *et al.* [1995]. Note that as no compensation is made for gravity, isenthalpic calculations should really only be performed at constant P , in which case the user controls the total number of iterations instead.

Finally, there is a phase diagram orientated mode that follows the limit of appearance of a particular phase or a line of constant melt fraction (by mass or volume) or activity of water in liquid as either pressure or temperature is incremented. This option may also be used to get suitable starting conditions for subsequent calculations in a different mode. For example, it can ensure that adiabatic decompression path begins at the solidus, for the given starting pressure. It can also be used like the 'Find Liquidus' menu option in the GUI version of MELTS e.g. for fractional crystallization. See menu options 3, 4 and 10 for more details.

Starting conditions

Initial P , T , entropy and / or fO_2 buffering may be set within one of the various input files (see below for more details) or adjusted manually via a menu option once the program is running. An initial guess for the phase assemblage is also required and this may be provided at the menu prompt within alphaMELTS or, optionally, in a text file. Two alternatives are given: for 'super-liquidus', a pure liquid trial solution is used and the algorithm progressively added the solid phases required [see Ghiorso, 1994]; for 'sub-solidus', the user must specify the solid phases to be used and the routine finds suitable modal proportions for the given composition [Asimow and Ghiorso, 1998]. The sub-solidus norm calculation works best for peridotites and fairly well for basalts; otherwise, if the

composition cannot be constituted from the given minerals, the program will default to a super-liquidus initial guess. Once started, the thermodynamic routines will add or drop phases as needed for geochemical equilibrium to be satisfied.

Trace elements and water partitioning

Routines for trace element partitioning between melt and solid phases were introduced in pHMELTS [Asimow, *et al.*, 2004]. In alphaMELTS we have extended the functionality to include isotopes (radiogenic or stable), user-defined partition coefficients and optional calculation of variable partition coefficients ($D = D(P,T,X)$) with the method of Wood and Blundy [1997]. The user specifies which trace elements to use in mineral-melt partitioning calculations, via the *melts_file*, and which phase-element pairs will have variable partition coefficients in the *trace_data_file*. Constant partition coefficients will be used for all other phases and elements. These can take default values from the compilations of McKenzie and O'Nions [1991; 1995] or be read in from the *trace_data_file*. We caution against using variable partition coefficients for any combinations not in the following table, as results may be unpredictable.

Table 1.

Phase	Reference	Allowed elements
Clinopyroxene	Wood and Blundy [1997] Blundy, <i>et al.</i> [1995] (for Na) Wood, <i>et al.</i> [1999]	Na, K, Sc, Ga, Rb, Sr, Y, Cd, In, Cs, Ba, La, Ce, Pr, Nd, Sm, Eu, Gd, Tb, Dy, Ho, Er, Tm, Yb, Lu, Ra, Ac
Garnet	van Westrenen <i>et al</i> [1999]	Sc, Y, In, La, Ce, Pr, Nd, Sm, Eu, Gd, Tb, Dy, Ho, Er, Tm, Yb, Lu
Feldspar (plagioclase)	Blundy and Wood [1994] (also see 'Trace Data file')	Li, Na, K, Rb, Sr, Y, Cs, Ba, La, Ce, Pr, Nd, Sm, Eu, Gd, Tb, Dy, Ho, Er, Tm, Yb, Lu

Originally, the solubility of water in olivine in pHMELTS was calculated using the Hirth and Kohlstedt [1996] model. Partition coefficients for water in orthopyroxene, clinopyroxene and garnet were linked to the olivine value using the mineral-mineral partition coefficients from Table 1 of Hirth and Kohlstedt [1996]; see Asimow, *et al.* [2004] for more details. The experiments of Mosenfelder, *et al.* [2005] established that water is incorporated into Fe-bearing olivines more readily than had previously been assumed so their expressions were adopted instead (e.g. see Hebert, *et al.* [2009]). The mineral-melt partition coefficients for opx, cpx and garnet may be either (a) as before (default), i.e. linked to the *olivine-melt* partition coefficient of Hirth and Kohlstedt [1996] value, or (b) linked to the higher water in olivine solubility, thus preserving the *mineral-mineral* values of Hirth and Kohlstedt [1996]. An option to use Hirth and Kohlstedt [1996] for olivine is retained and it is also possible to independently increase the solubility of water in opx to match that in cpx, in keeping with the results of Hauri, *et al.* [2006]. See the 'pHMELTS and Trace Elements' section on environment variables for more details.

Open versus closed system behavior

As mentioned in the introduction melting may be batch, fractional or continuous, where some but not all of the melt is extracted. The amount of melt retained in the last case may be a constant mass fraction, a constant volume fraction (which can be thought of as the residual porosity along grain boundaries etc.), or a constant ratio of the available melt. Alternatively, the amount of melt extracted may be adjusted at each step so as to maintain a constant volume. Similarly, crystallization may be

batch or fractional and various options exist for the latter. By default, a nominal (adjustable) amount of each solid phase is retained during fractionation, mainly to stabilize the calculation. Selective fractionation of solid phases can be useful, for example, to simulate the isolation of some phases in the magma chamber due to buoyancy effects. It is also possible to fractionate the water phase in an analogous way to the extraction of melt during continuous melting. See the ‘[Open vs. closed system](#)’ settings for more information; for selective fractional crystallization see the ‘[MELTS file](#)’ section.

For calculation modes in which the PT -path is imposed, such as isobaric and geothermal, a variety of fO_2 constraints are available (see ‘[MELTS file](#)’ section) and a selection of options can adjust the behavior of the fO_2 buffer or aid program stability when fO_2 -buffered calculations are relatively difficult (e.g. sub-solidus or for particular bulk compositions; see ‘[P-T path and fO2 control](#)’ section on environment variables). As currently formulated, however, it is not possible for calculations that fix entropy, enthalpy or volume to be simultaneously fO_2 -buffered as, say, this might imply that oxygen entering or leaving the system carried no heat. Instead, there is an option to approximate fO_2 buffering in isentropic, isenthalpic and isochoric modes. The fO_2 buffer is switched off while the new T or P is solved for. Then an extra equilibration step reestablishes the fO_2 buffer for the current P , T conditions and adjusts the reference entropy, enthalpy or volume for any net change in oxygen content. This alternation between closed and open system behavior is comparable to the way melt extraction is handled in most geochemical models of adiabatic (‘isentropic’) decompression melting, including alphaMELTS.

There is also an option to buffer the water activity. For water saturated (p)MELTS calculations the user will set the water content in wt % and normally (i.e. no H_2O buffering) the water content will be maintained during the calculation. Alternatively, the pHMELTS model provides an option to treat H_2O as a trace element and in this case the total H_2O in the liquid, hydrous and nominally anhydrous solids will normally be fixed at the specified value as the calculation proceeds. However, if aH_2O is fixed to some value $0 < aH_2O \leq 1$ in MELTS or pMELTS then total water content will be adjusted during the calculation to give the desired activity of water in the melt. This does not work along an isentrope, isenthalpic path or isochore and at present, unlike fO_2 , there is no approximation for H_2O buffering along such paths.

Special calculations

For adiabatic decompression calculations, the output from alphaMELTS includes melt fractions and compositions for each pressure. If melting is fractional or continuous the melt increments may be integrated for 1-D (column) and 2-D (triangular) melting regimes [Asimow, *et al.*, 2001]. The *integrated_output_file* generated contains the thickness, composition and mean properties of the crust, as well as aggregated liquids at all depths. An intermediate file can be written during melting calculations in case the program crashes and then used to generate *integrated_output_file* output by restarting alphaMELTS (see ALPHAMELTS_INTEGRATE_FILE and option 16 for more details). If melting is fractional or continuous but performing the pressure integral is inappropriate (e.g. for isobaric heating) a simple weighted sum of the extracted melts may be output instead.

We have incorporated and extended the fractionation correction schemes of Asimow, *et al.* [2004] and Cooper, *et al.* [2004], including the ‘Amoeba’ routine that, given an initial estimate, varies the parental melt composition until isobaric forward fractionation yields a specified target. The alphaMELTS version is flexible in the choice of varied oxides and fO_2 constraint, and offers optional fitting to the liquid line of descent (LLD) of a whole suite (see Antoshechkina, *et al.* [2010]).

It is also possible to reproduce the melt focusing calculation described in Asimow and Stolper [1999]. This option multiplies the mass of liquid in the system by a fixed factor after each equilibration in

order to model the migration of melt in channels or simulate two-dimensional flow with converging melt streamlines. One further option allows major element compositions to be calculated for systems with multiple liquids, although trace element partitioning is then unavailable, and another option can be used for constructing ‘metastable inverse diagrams’ like those in *Asimow and Longhi*, [2004].

Output

Output includes modes and compositions for all solid and liquid phases, as well as thermodynamic data and density and viscosity estimates. Most information is dumped at every step, in case the program crashes; the resulting output may also be combined into a single file, name chosen by the user, at the end of the run. Other output, such as aggregate melt compositions or more detailed phase thermodynamic data, may be written by choosing appropriate menu options.

Importing many different files into spreadsheet programs, such as Excel, may be time consuming. The Perl script `column_pick.command` is a robust way to generate one space, comma or tab-delimited text file containing the user’s choice of columns from as many files as desired. With suitable settings within Excel, the new file can be easily imported into a single worksheet. An example Matlab function is also provided that can be used to call the script and process the output, without the need for the intermediate text file. Although not strictly necessary, if you wish to use `column_pick.command` we recommend you stick to the `run_alphamelts.command` / `alphamelts` convention of using ‘_’ instead of spaces in file names (but not folders).

How to cite alphaMELTS

A full bibliography is given at the end of this document, which includes references that should be cited for source compositions and partition coefficients that are included as part of the program, as well as alternative solid solution models for garnet, spinel and biotite.

The original `Adiabat_1ph` front end, which alphaMELTS builds on, was described in:

Smith, P. M., and P. D. Asimow (2005), *Adiabat_1ph: A new public front-end to the MELTS, pMELTS, and pHMELTS models, *Geochem. Geophys. Geosyst.*, 6, art. no. Q02004, doi:10.1029/2004GC000816.*

It is important that you also cite the appropriate thermodynamic models. For calculations performed in MELTS mode (including pHMELTS crystallization) use:

Ghiorso, M.S., and R.O. Sack (1995), *Chemical Mass-Transfer in Magmatic Processes IV. A Revised and Internally Consistent Thermodynamic Model for the Interpolation and Extrapolation of Liquid-Solid Equilibria in Magmatic Systems at Elevated-Temperatures and Pressures, *Contributions to Mineralogy and Petrology*, 119 (2-3), 197-212.*

Asimow, P.D., and M.S. Ghiorso (1998), *Algorithmic modifications extending MELTS to calculate subsolidus phase relations, *American Mineralogist*, 83 (9-10), 1127-1132.*

If calculations are performed in pMELTS mode (including pHMELTS melting) use:

Ghiorso, M.S., M.M. Hirschmann, P.W. Reiners, and V.C. Kress (2002), *The pMELTS: A revision of MELTS for improved calculation of phase relations and major element partitioning related to partial melting of the mantle to 3 GPa, *Geochemistry Geophysics Geosystems*, 3(5), art. no. 1030, doi:10.1029/2001GC000217.*

For pHMELTS calculations, in addition to the above, please reference:

Asimow, P.D., J.E. Dixon, and C.H. Langmuir (2004), A hydrous melting and fractionation model for mid-ocean ridge basalts: Application to the Mid-Atlantic Ridge near the Azores, *Geochemistry Geophysics Geosystems*, 5, art. no. Q01E16, doi:10.1029/2003GC000568.

The most significant additions to the alphaMELTS front end since the original *Adiabat_1ph* release are described in the following, which may supplement *Smith and Asimow* [2005]:

Thompson, R. N., A. J. V. Riches, P. M. Antoshechkina, D. G. Pearson, G. M. Nowell, C. J. Ottley, A. P. Dickin, V. L. Hards, A.-K. Nguno and V. Niku-Paavola (2007), Origin of CFB magmatism: Multi-tiered intracrustal picrite-rhyolite magmatic plumbing at Spitzkoppe, western Namibia, during early-Cretaceous Etendeka magmatism, *J. Petrol.*, 48, 1119-1154.

Antoshechkina, P. M., and P. D. Asimow (2010), alphaMELTS 3.0 and the MAGMA website: educational and research tools for studying the petrology and geochemistry of plate margins, Abstract ED41B-0644 presented at 2010 Fall Meeting, AGU, San Francisco, Calif., 13-17 Dec.

Antoshechkina, P. M., P. D. Asimow, E. H. Hauri, and P. I. Luffi (2010), Effect of water on mantle melting and magma differentiation, as modeled using alphaMELTS 3.0, Abstract V53C-2264 presented at 2010 Fall Meeting, AGU, San Francisco, Calif., 13-17 Dec.

Setting up alphaMELTS

Before you start

In this documentation we use the following conventions: `courier` font plain text for executable files and commands that must be typed in and italics to denote a variable, such as a file name, that should be substituted into the command line. A phrase like ‘`alphamelts` menu prompt’ is used to refer to user typed input (as opposed to input read in from a file) while the `alphamelts` executable is running and ‘command line’ for things typed at the terminal before or after `alphamelts` is run.

The `alphamelts` or `alphamelts.exe` executable is invoked by using the Perl script `run_alphamelts.command`, which in turn requires that the Perl program be installed on the computer. Although it is possible to use `alphamelts` as is (i.e. without Perl, or without following the installation procedure / setting the path) we do not support this mode of interaction. *Note also that ‘double-clicking’ the `alphamelts` executable will not generally result in correct behavior.*

Perl should be included on Linux, Mac OS X and also Windows machines that have Matlab installed. Perl is also part of the Windows Subsystem for Linux (WSL), which is included in the 64-bit version of Windows 10. Enabling the WSL requires one-off administrator access, but provides what is arguably the smoothest workflow for using alphaMELTS (and also the Rhyolite-MELTS GUI) on Windows. A detailed comparison of WSL alphaMELTS, VM alphaMELTS and native Windows alphaMELTS is given on the alphaMELTS download page, together with instructions on how to enable the WSL on 64-bit Windows 10 and/or install VM alphaMELTS on other systems.

If Perl is not already installed, go to <http://www.perl.org/get.html> and follow the appropriate link for your operating system. For Windows users who do not have access to an administrator account, we recommend you download and extract the latest ‘portable’ version of Strawberry Perl from

<http://strawberryperl.com/releases.html>. Other Windows users may prefer to install ActivePerl, which is available at <http://www.activestate.com/activeperl/downloads>.

alphaMELTS version 1.5 onwards has (very useful) tab-completion within the `alphamelts` menu, if Richard Kettlewell's `with-readline` program is installed. The `with-readline` program is distributed under the terms of the [Gnu Public License](#). Ideally, `with-readline` should be installed before alphaMELTS (though it can be added later). On Mac, the easiest way to install `with-readline` is via [Homebrew](#). For Ubuntu (including the Windows Subsystem for Linux, which is based on Ubuntu by default) `.deb` packages are provided in the alphaMELTS download. For other types of Linux, such as CentOS, it is fairly simple to build the `with-readline` program from source. More details and links are available on the operating system-specific boards of the forum.

Once the alphaMELTS package has been installed, the most powerful way to use it is to call `run_alphamelts.command` from a terminal program. In Mac OS X the 'Terminal' program is available in 'Applications/Utilities' within 'Finder'. In Windows, a similar program called 'Command Prompt' can be found quickly by clicking the Start Menu (Windows 7) and typing 'cmd.exe' in the search box, or by right-clicking the Start Button (Windows 10), choosing 'Run' and typing 'cmd.exe' there. It will help to turn on 'Quick Edit Mode' by right-clicking the `cmd.exe` window header and choosing 'Properties' from the context menu. Depending on the exact flavor of Linux being run, the 'Terminal', 'Terminal (Konsole)' or 'Terminal Emulator' program may be located in the 'Accessories', 'System Tools', 'System' or 'Installed' submenu of the 'Applications' menu. Linux and Mac users should use the 'bash' shell when invoking any of the '.command' scripts from the terminal command line. On most systems 'bash' will be the default shell, but if it is not then type 'bash -l' (that is lowercase 'L', not 'one') and then press enter before continuing.

All of the '.command' scripts within the alphaMELTS package can also be used by double-clicking, which will open the appropriate terminal program automatically; files may be dragged and dropped from Mac's 'Finder', 'Windows Explorer' or a Linux file browser, such as 'Nautilus'. This alternative mode of operation simplifies the process of navigating around directories and managing files and also side-steps some of the platform dependent set up. It is therefore particularly useful for teaching exercises, quick *ad-hoc* calculations, or for users whose accounts have limited privileges.

Installation

The procedure described here should work for most set-ups. Brief installation instructions are given on the alphaMELTS download site and alternative strategies are outlined elsewhere. Check out the 'Windows', 'Linux' and 'Mac OS X' boards under 'Operating system specific' on the forum, though note that some issues are posted there because they are more common on a particular operating system (e.g. line-ending problems mostly occur on Macs) but are not necessarily exclusive to it. In particular, Linux users may find useful information on the Mac OS X board and vice versa. If you cannot get the program or scripts to work after consulting the documentation and forum please post a query or contact us (at psmith@gps.caltech.edu) and we will try to find a solution.

alphaMELTS software packages for all systems are provided as `.zip` files. It should be possible to extract the files by double-clicking the archive. A simple script `install.command` is provided which makes shortcuts to the scripts and program and then tries to ensure that the folder containing the shortcuts is in the user's 'PATH' environment variable. The 'PATH' variable lists the places where the system will look for executable programs and scripts. The install script also makes sure that the line endings in the scripts and example files are correct for the operating system it is being run on. The files from the software download site all have Windows' line-endings.

By default the shortcuts that `install.command` makes point to the scripts and program in the original *download_folder*. If you select a different location, the script will try to copy the `alphamelts` executable and the `.command` scripts (except `install.command` itself) to this *installation_folder* and link the shortcuts to the copies. Normally the installation script places the shortcuts in a folder called `bin` in the user's home space or, on Windows, in the user's "Documents"¹ folder. Another more convenient location (or locations) may be chosen, e.g. on the Desktop or near the user's input files. If more than one *links_folder* is made, by rerunning `install.command` or by copying and pasting the *links_folder*, only one need be in the PATH².

We strongly recommend using a separate folder for any input files (i.e. not the *links_folder*, *installation_folder*, or *download_folder*, although a subfolder is fine) and calling the `.command` scripts from there. Later in the documentation we sometime call this location the `'working folder'` (see ['Interactive operation'](#)). As a start, the install script will make copies of all the example input files provided in the alphaMELTS package. By default it will place these in a folder called `'examples'` but this can be moved or renamed later.

- ***Standard installation on Mac or Linux (includes Windows Subsystem for Linux):*** For the Windows Subsystem for Linux (WSL) you will need to open the bash terminal, navigate to the *download_folder* navigate and type `./install.command`. Mac and Linux users should be able to double-click `install.command` but it can also be called from the command line. Once open, the script will test for a previous installation and if it detects one, print a message asking whether to continue installation.

The script will then ask for the *installation_folder*, *links_folder*, and *examples_folder* locations. Press `'return'` to use the suggested folder, or type or drag-and-drop a folder name with its complete path (see ['Interactive operation'](#) for more information). Finally, it will ask for the default name (and path) for the `'Settings file'`; for most purposes, especially first time installation, `'alphamelts_default_env.txt'` will be fine. The script generates quite a lot of output but it is mostly for information only. Certain environment variables or files will not be altered; instead the script will indicate what may need to be modified manually. On Linux and Mac OS X, files beginning with a dot, such as `'profile'`, are hidden files; consult the forum if you are not sure how to edit these. When you are done follow any instructions e.g. to log out and in. When you double-click the install script again you should find that the path has been set and can choose `'n'` to abort another installation.

- ***Advanced installation on Linux or Mac (includes WSL):*** This is useful if you want to use put the *installation_folder* and / or *links_folder* in a location where the normal user does not have write-permission e.g. `'/usr/local'`. To run as `'root'` on Linux or Mac, if allowed, you must run the script from the command line. Open the terminal program and navigate to the *download_folder*. Type `'sudo ./install.command'` and enter your user password. Note that for Windows 10, this is the password you set up when enabling the WSL and not your Windows password. Please do not run the script with elevated privileges unless you are

¹ Although the containing folder is called "Documents" on disk, Windows Explorer will display the name in the chosen language for the operating system e.g. "Dokumente" for German, "Documentos" for Spanish.

² If the script is being run using `'sudo'` on Mac or Linux then it will not attempt the PATH set-up step... so it is possible, for example, to run the script twice: once with `'sudo'` to put the executables and links in, say, `'/usr/local/MELTS'` and `'/usr/local/bin'` respectively and then a second time, without `'sudo'` but using the existing *installation_folder*, to make a *links_folder* for the normal user (you can ignore the copy error that this generates).

confident that you know what you are doing; we will not be responsible for the consequences. It is a good idea to run the script as the normal user first to see how it behaves.

- ***First time installation on Windows (except Windows Subsystem for Linux):*** The very first time they use the install script, Windows users will need to right-click on the `install.command` file to select ‘Open with...’ (or ‘Open’, if ‘Open with...’ is not available). Choose Perl, or if it is not listed as a recommended program then browse to the `perl.exe` file³. The install script will then ensure that, for the current user, all files with the ‘.command’ extension are permanently assigned to the chosen Perl executable. You can verify that it has worked by closing the install script when it is finished (see below) and then double-clicking it to open it again.

Once open, the script will ask for the *installation_folder*, *links_folder*, and *examples_folder* locations. Press ‘return’ to use the suggested folder, or type or drag-and-drop a folder name with its complete path (see ‘[Interactive operation](#)’ for more information). Finally, it will ask for the default name (and path) for the ‘[Settings file](#)’; for most purposes, especially first time installation, ‘`alphamelts_default_env.txt`’ will be fine. The script generates quite a lot of output, particularly on Windows, but it is mostly for information only. The script will back up any Windows registry entries it alters as ‘.reg’ files (assuming the installation is successful these may be removed afterwards). Certain environment variables or files will not be altered; instead the script will indicate what may need to be modified manually. Brief instructions will be printed; note that you may need to scroll up slightly to see the whole message if the path to your *links_folder* is very long. When you are done follow any instructions e.g. to log out and in. When you double-click the install script again you should find that the path has been set and can choose ‘n’ to abort another installation.

- ***Subsequent installation or update on Windows (except WSL):*** When rerunning the install script (e.g. to add a second *links_folder* or when installing a later version of alphaMELTS) it should be possible to double-click `install.command` to open it. Once open, the script will test for a previous installation and if it detects one, print a message asking whether to continue installation. The script will ask you to choose the *installation_folder*, *links_folder*, and *examples_folder* locations, as before (see previous paragraph). `install.command` can also be called from the command line: open the terminal program, navigate to the *download_folder* and type ‘`install.command`’.
- ***Advanced installation on Windows (except WSL):*** This is useful if you want to use put the *installation_folder* and / or *links_folder* in a location where the normal user does not have write-permission e.g. “C:\Program Files”. To run with elevated permissions on Windows you must actually open `cmd.exe` as an administrator. To locate the ‘Command Prompt’: click (Windows 7) or right-click (Windows 10) the Start Menu button and type ‘`cmd.exe`’ in the search box. Right-click the ‘Command Prompt’ or `cmd.exe` result and choose ‘Run as

³ The Strawberry Perl executable will usually be at ‘`strawberry_perl_folder\perl\bin\perl.exe`’; the Matlab version will be something like ‘`C:\Program Files\Matlab\Rrelease\sys\perl\win32\bin\perl.exe`’, where *release* is the year followed by ‘a’ or ‘b’; and the ActivePerl one should be at ‘`activeperl_folder\bin\perl.exe`’. Note that if a Perl program has been installed on the Windows machine previously (e.g. to use `Adiabat_1ph 2`) then that version should be used. If you wish to use the `perl.exe` included in Matlab or the ‘portable’ version of Strawberry Perl instead, then you need to uninstall the other Perl program first to avoid conflicts. Note also that the ‘Open with...’ association need only be set up once (for each user). This step can be omitted when installing subsequent releases of the alphaMELTS package.

administrator’ (you may need to do this even if you are logged into an administrator account). Please do not run the script with elevated privileges unless you are confident that you know what you are doing; we will not be responsible for the consequences. It is a good idea to run the script as the normal user first to see how it behaves.

It is important that the *links_folder* is in the user’s path if alphaMELTS is to behave correctly in all the ways it is supposed to (see [‘Interactive operation’](#)). `install.command` will determine if the path is set and will print brief instructions only if necessary. If you accidentally shut the window too soon then run `install.command` again using the same folder choices. The operating-system-specific boards on the forum have more information on setting the path, including workarounds if this is not possible.

When there is a new release of the alphaMELTS package the installation procedure can be repeated – there is generally no need to uninstall the older version. If you would like to change the location of the *installation_folder* or *links_folder* at any time then you can simply rerun `install.command`. This will leave the old installation in place, however, so you may want to delete files from the old *installation_folder* and / or *links_folder* manually. You may also need to remove the old *links_folder* from your PATH (see the forum for more details).

From time to time we release updates to the `alphamelts` executable alone. If an update is available at the time of installation then download the ‘.bin’ file and place the file in the *download_folder* before running `install.command`. Otherwise, we provide a pared down version of the install script for updating. Just put the downloaded ‘.bin’ file in the same folder as `update.command`. Follow the instructions above to run `update.command` in the same way as you did `install.command`. It will rename the `alphamelts` executable and ensure that the shortcut points to it correctly. Similarly, if you install `with-readline` after installing alphaMELTS, then run `update.command` to make sure that the `with-readline` program is detected.

Interactive operation

The Perl script `file_format.command` can be used to process any text file in order to ensure that it has the correct line-endings for the system it is being used on. Incorrect line-endings in input files could produce seemingly random errors in early versions of `adiabat_lph/run_adiabat.pl`. Although we have mostly fixed the problems with line-endings we provide the script, just in case (it is also used by `install.command`). Here we use it as a simple example to illustrate the different ways in which all the Perl scripts in the alphaMELTS package (e.g. `run_alphamelts.command`, `column_pick.command`) can be invoked.

For the sake of this example the *examples_folder* will be the ‘working folder’ (see [‘Installation’](#) for more information) and we will run the script from there. Once the terminal is open, navigate to the *examples_folder* using one or more ‘`cd folder`’ commands. Then type:

```
file_format.command melts_files
```

where *melts_files* is a list of the ‘.melts’ files separated by spaces. *Note that there should be no ‘perl’ or ‘./’ before ‘file_format.command’, even on Linux or Mac OS X.* If the installation procedure was successful you should see a list of ‘Processed *melts_file*’ messages, one for each file. On most systems you can get the same effect by using ‘*’ as a wildcard, i.e.:

```
file_format.command *.melts
```

though this second format is of less use for the other ‘.command’ scripts. If you are typing a long file or folder name then pressing the ‘tab’ key after typing some of the name will (somewhat) complete the name automatically and help to make sure any special characters are correctly escaped or quoted. Note, however, that the behavior of ‘tab’ on Windows is slightly different to Linux / Mac OS X: the former will take the first alphabetical match (e.g. ‘script.command’ when choosing between ‘script.command’ and ‘script.pl’) whereas the latter will partially complete the name and wait for more input (e.g. ‘script.’, at which point you can type ‘c’ or ‘p’ followed by ‘tab’; if the ‘deciding character’ is a special one, e.g. a space, you may need to type ‘\’ before it).

You can also call the ‘.command’ scripts without command line options. If you type simply ‘file_format.command’ you will get a message outlining the script usage and will then be prompted for the list of *melts_files*⁴. After these files have been processed you can press ‘y’ or ‘n’ to process more files. It is also possible to replace all the command line input, in this case the list of *melts_files*, with a text file. This is rarely needed for *file_format.command* but can be useful when a long list of options is passed to *run_alphamelts.command* (see ‘[Command line switches](#)’). For example, you could type:

```
file_format.command < melts_file_list
```

where *melts_file_list* is a text file containing the following:

```
McKenzie_ONions_DM.melts McKenzie_ONions_PM.melts
y
Morb.melts
y
Sun_McDonough_PM.melts Workman_Hart_DMM.melts
N
```

To use the scripts in double-click mode, open the *links_folder* using a file manager, such as ‘Finder’ or ‘Windows Explorer’. Open a second window of the file manager and navigate to the *examples_folder*. Next, double-click the *file_format.command* shortcut (you may be asked to confirm that you do indeed wish to run the script in a terminal). On Windows the ‘shortcut’ is actually a file called ‘*file_format.command.bat*’ but the ‘.bat’ extension may or may not be displayed. The terminal program will open and display a message outlining the script usage. The user will then be prompted for the list of *melts_files*. Bear in mind that the script is now running in the *links_folder*⁵ and not in the *examples_folder* so typing ‘Morb.melts’, say, will generate an error. Instead click the icon for the ‘Morb.melts’ file in the file manager window and drag it onto the terminal window. A little plus sign should appear on the file icon and when you release the mouse button the ‘Morb.melts’ file name, with its complete path, will have been added to the command line. You may need to click the window to bring it back into focus. Then press ‘return’ and you should get a message saying the file has been processed successfully.

On Linux and Mac you can actually select several files to drag and the system will produce a space-delimited list of file names. On Windows the files must be dragged individually and the user must press the space key inbetween. Alternatively, open a text editor and drag the files to that; save the file

⁴ As it is the Perl script, rather than the system, that processes the filenames the ‘*’ wildcard will not work in this case.

⁵ On Mac OS X the scripts run in the user’s home space when they are double-clicked, rather than in the *links_folder*. Either way, putting the full path by dragging-and-dropping a file or folder is by far the easiest way to ensure it is found.

as *melts_file_list*; then when `file_format.command` is open type ‘<’ and drag-and-drop *melts_file_list* onto the terminal window. Note that the system will escape awkward characters (e.g. with ‘\’) and / or put the full filename in quotes. Except where stated otherwise, the `alphamelts` executable, `file_format.command` and its sister scripts are all designed to cope with most of the special characters that might *reasonably* appear in file and path names, such as spaces, apostrophes and parentheses. More than one consecutive space is not allowed and there could be other exceptions (please let us know if you find one). You can also drag-and-drop files onto the terminal command line when using the ‘.command’ scripts in the normal mode(s) of interaction described above.

Finally, it is possible to run the Perl scripts from within Matlab. First set the Matlab path variable to include the *links_folder* (type ‘help path’ at the Matlab prompt). Alternatively, rerun `install.command` with the same *installation_folder* as before and use the existing Matlab user folder as the *links_folder* (type ‘userpath’ to find the location). When using the ‘perl’ function the first argument needs to be the name of the script and subsequent arguments are the command line arguments that would be separated by spaces in normal mode. For example, the following could be typed at the Matlab prompt or included in a ‘.m’ script:

```
message = perl('file_format.command', 'Morb.melts', ...  
              'Sun_McDonough_PM.melts', 'Workman_Hart_DMM.melts')
```

If Matlab is running in a different folder to the *examples_folder* then the path to the *examples_folder* must be included in the filenames. It is possible to drag-and-drop a text file from Matlab’s ‘Current folder’ to its ‘Command Window’; this opens the file using ‘uiopen’ and shows how to put the full filename into Matlab’s string format (i.e. contained by single quotes, with apostrophes escaped by putting two consecutively). The screen output (STDOUT) and error messages (STDERR) from the script and, if appropriate, the `alphamelts` program are returned in the ‘message’ variable. This method of calling from Matlab is simplistic – a fully integrated alphaMELTS library and Matlab interface, where individual menu options can be called from Matlab and phase compositions, thermodynamic properties etc. are returned as variables (rather than written to text files) is work in progress....

Using alphaMELTS

Synopsis

```
run_alphamelts.command [-h] [-p output_path] [-f settings_file] [-b  
  batch_file | -a] [-m melts_file] [-o output_file] [-l  
  log_file]
```

Square brackets surround each optional part in the command line and the pipe (‘|’) denotes ‘or’. (Do not actually type ‘[’, ‘]’ or ‘|’.) Each option is described below in ‘Switches’. The order of the switches should not matter except that ‘-h’ (‘help!’) should be used alone and will be ignored if it follows any others. Typing ‘run_alphamelts.command -h’ will give a brief description of the various command line options; otherwise `run_alphamelts.command` will work through your chosen switches, writing files and setting environment variables as appropriate, before starting the main `alphamelts` program.

Note that in double-click mode the user will generally need to make an *output_path* folder and use it to set the ‘-p’ switch, else the output files will end up in the *links_folder*. If calling the script from Matlab, the user will need to use a *batch_file* with the ‘-b’ switch as the screen output (including the

alphamelts menu prompt) is not returned to Matlab until the alphamelts program and run_alphamelts.command script have both finished.

Input file formats

Note that none of the input files are case sensitive but it is important that there are only single spaces in the *melts_file*, *enrichment_file* and *trace_data_file* (except at the end of the line). Completely blank lines will be ignored but lines that contain only white space may generate an error. Mac users, in particular, and anyone transferring input files from one operating system to another may need to run *file_format.command* on the input files, as described above.

MELTS file

Input files for the GUI version of MELTS and pMELTS can be used in alphaMELTS, except that some lines (e.g. those beginning 'Increment' or 'Final') may be ignored as the *P-T* path can be controlled with environment variables instead. We also introduce the 'Initial Entropy:', 'Initial Enthalpy:' and 'Initial Volume:' lines (in the GUI version these reference quantities are always calculated for the initial *P-T* conditions specified) and the 'Initial Mass:' line, which is used to scale the system so that total oxides sum to the given number of grams. The 'Initial Trace:' line is also specific to alphaMELTS. All valid lines are shown in the example below, where the 'Title:' and 'Initial Composition:' lines are the only compulsory ones (see the note about ['Note about initial P, T, etc.'](#)). 'Title:' should be first; otherwise the order should not matter. There may be multiple 'Title:' lines in a given file, providing a mechanism to add comments, but only the last one will be printed in any output.

Masses of major oxides (possibly including H₂O) are in grams; a zero entry will ensure the oxide appears in any screen or file output. Temperature is in °C, pressure in bars and entropy in J K⁻¹. The possible settings for the 'Log fO₂ path:' are FMQ, integer values -9FMQ to +3FMQ, IW, NNO, HM, ABS (or ABSOLUTE), or NONE. 'Log fO₂ Delta' is alphaMELTS-specific, is expressed in log units and may optionally be used to offset from the chosen buffer or to impose an absolute value; there is no limit on the offset, which need not be an integer. The first two 'Log fO₂' lines in the example are equivalent to a single 'Log fO₂ Path: -1FMQ' line. Note that the GUI version of Rhyolite-MELTS will always include a pressure term, when calculating the FMQ buffer; alphaMELTS does not, by default, but can be made to do so using the ALPHAMELTS_FO₂_PRESSURE_TERM environment variable. If *fO₂* buffering is selected and you are using a superliquidus start you may omit the 'Initial Composition:' for Fe₂O₃. For *fO₂* buffering with a subsolidus start you must put some of the Fe as Fe₂O₃ for the initial guess routine to behave correctly but the exact proportions of FeO and Fe₂O₃ in the *melts_file* are not important, as the program will adjust them.

```
Title: title_goes_first
Initial Composition: SiO2 45.06
Initial Composition: TiO2 1.04
Initial Composition: H2O 1.0
    ... list all oxides to be included ...
Initial Trace: La 0.687
Initial Trace: Lu 0.074
Initial Trace: H2O 200.0
    ... list all trace elements in the order you would like them to be output ...
```

... continued from previous page ...

Initial Trace: Nd 1.354

Initial Trace: Nd143 0.33917

Initial Trace: Nd144 0.66083

Initial Temperature: 1100.0

Initial Pressure: 15000.0

Initial Mass: 50.0

Initial Entropy: 260.0

... or Enthalpy or Volume ...

Log fO2 Path: FMQ

Log fO2 Delta: -1.0

Suppress: liquid

Suppress: quartz

... repeat as many times as needed to turn off unwanted phases ...

Limit coexisting: orthopyroxene

Limit coexisting: clinopyroxene 2

Limit coexisting: feldspar 0

... repeat as many times as needed to limit exsolution in (solid) solution phases ...

Fractionate: quartz

Fractionate: feldspar 1

... repeat as many times as needed to fractionate phases selectively...

The following lines, which come from the rhyolite-MELTS GUI and alphaMELTS 2, are only used in alphaMELTS 1.9 and not in version 1.8.1 or earlier. Previously these lines would be ignored:

Log fO2 Path: +3FMQ

... note the integer must be before the buffer, with no gap ...

Mode: isenthalpic

Mode: isentropic

Mode: isochoric

As above, except that alphaMELTS 1.9 will still ignore these lines unless the '-m melts_file' switch is used with run_alphamelts.command (see below). Note that setting the relevant environment variables will override these values, so that alphaMELTS 1.9 is backwards or forwards compatible:

Final Temperature: 800.0

Final Pressure: 500.0

Increment Temperature: 5

Increment Pressure: 1000

... sign for Increment determined by corresponding Initial and Final values ...

Mode: fractionate liquids

Mode: fractionate fluids

Mode: fractionate solids

Mode: multiple liquids

The trace elements should be listed in the order in which you would like them to appear in the output file (see below) and can include elements also listed as majors. They may be input as ppm (except isotopes) or as normalized ratios. For example, to normalize to the source composition, simply list each trace element in order with a '1.0' (or a fraction for isotopes). The program checks that each element listed actually exists in the periodic table. There is an option within the program to normalize concentrations to one of four published source compositions (see below). There is a sample *melts_file* for each of these four source compositions. Fe is distributed using the rule $\text{FeO}^* = \text{FeO} + 0.9 \text{Fe}_2\text{O}_3$ in these example files. If $f\text{O}_2$ buffering is not selected it may be desirable to alter the distribution of Fe between FeO and Fe_2O_3 manually.

As described below, to use the pHMELTS algorithm and treat water as a trace element the environment variable ALPHAMELTS_DO_TRACE_H2O must be set. H₂O content may be input in the *melts_file* as 'Initial Composition: H2O:' in grams and / or 'Initial Trace: H2O' in ppm. The total water will be summed but only water input as grams of oxide will be included in any scaling of the system using 'Initial Mass:' or if a reference quantity, such as entropy, is set (e.g. by an 'Initial Entropy:' line or if option 7 is called before any calculations). Alternatively, the water content may be input or modified at the menu prompt in `alphamelts` (in ppm or, if water buffering is chosen, as *aH2O*); unless it is already included, 'H2O' will be added to the trace element list.

If ALPHAMELTS_DO_TRACE_H2O is not set then water input as 'Initial Composition: H2O' will be handled in the normal way for MELTS or pMELTS, which means in practice that it will usually all go into the melt. If water is input as 'Initial Trace: H2O' it will be partitioned between melt and solids but will have *no effect on melting or phase equilibria*. There will be no exchange of water between the major element budget, which is handled by the (p)MELTS algorithm and subject to thermodynamic controls on melting and the abundance of hydrous minerals, and the trace element reservoir that includes the nominally anhydrous minerals and is tracked by the trace element routines [Asimow, *et al.*, 2004]. We do not intend trace element partitioning of water to be used without setting ALPHAMELTS_DO_TRACE_H2O but it will not in itself generate an error.

Note that MELTS is calibrated in the system SiO₂-TiO₂-Al₂O₃-Fe₂O₃-Cr₂O₃-FeO-MnO-MgO-NiO-CoO-CaO-Na₂O-K₂O-P₂O₅-H₂O. pMELTS is calibrated in the system SiO₂-TiO₂-Al₂O₃-Fe₂O₃-Cr₂O₃-FeO-MgO-CaO-Na₂O-K₂O-P₂O₅-H₂O. The effect of including any other components (i.e. CO₂, SO₃, Cl₂O, F₂O, and, in pMELTS, NiO, MnO, and CoO), though it will not cause the program to crash while reading in the *melts_file*, is unpredictable and is not recommended. Such elements may be entered as trace elements instead (though note that CO₂, S, Cl and F are not included in the compilation of McKenzie and O'Nions [1991; 1995] so their partition coefficients must be defined manually or default to zero; see the 'Trace Data file' description below). Furthermore, the subsolidus routines do not handle K₂O well if feldspar is not present, so calculations at subsolidus conditions will be more successful (and more informative) if K is also treated as a trace element. It may help to remove K₂O and / or Cr₂O₃ lines in other circumstances too (see the Cr₂O₃ note in 'Bugs and Fixes').

For the reasons stated above, some elements (i.e. Ni, Cr, Mn and K) appear as both major and trace elements in the examples files. Any NiO, MnO and CoO lines should be removed for pMELTS calculations. Elements that have both 'Initial Composition: *oxide*' and 'Initial Trace: *element*' lines in the same file will behave in a similar manner to H₂O with ALPHAMELTS_DO_TRACE_H2O unset i.e. the two entries are duplicates that are used separately by the thermodynamic and trace element partitioning parts of the code and the concentrations will never be summed.

Any real element may be entered, together with the species 'H2O' and 'CO2' (see above). Element names beginning with 'Q' and followed by another letter (i.e. 'Qa', 'Qg', 'Qx'...) may be employed

for user-defined species e.g. a perfectly compatible or perfectly incompatible one. In addition all element names may be followed by up to three digits so that isotopes (e.g. 'Li6', 'Li7', 'Nd143', 'Nd144', 'Pb208', 'Pb207', 'Pb206', 'Pb204'...) or elements with multiple valencies (e.g. Eu2, Eu3) can be used. It is up to the user to ensure that the input isotopes actually exist. Radiogenic isotopes will be useful if the source mixing, assimilation or flux melting functions are used, whereas stable isotopes can be given different partition coefficients (see ['Trace Data file'](#) below) and used to model isotopic fractionations. Isotopes should appear after the 'non-isotopic' version of the element as the normalized composition will be *multiplied* by the total concentration for that element. The example shown above is for $^{143}\text{Nd} / ^{144}\text{Nd} = 0.51325$ and the stored source for ^{143}Nd will be 0.459 ppm.

By default, all solid and liquid phases are 'switched on', meaning the algorithm will test for their saturation during equilibration. As mentioned previously, a full list of the solid phases in MELTS, pMELTS and pHMELTS is available on the forum. Phases may be 'switched off' completely by adding a 'Suppress: phase' line or the maximum number of instances imposed with a 'Limit coexisting: phase' line. If no value is given then at most one instance of a phase will be stabilized; if a value of '0' is given, this is exactly equivalent to using a 'Suppress: phase' line. By contrast, liquid is included on the list of potential phases but limited to a single instance by default. There is an environment variable, ALPHAMELTS_MULTIPLE_LIQUIDS, that can be used for calculations in areas of composition space for which liquid miscibility may be important.

Fractional crystallization of all solid phases (except water) is turned on with the environment variable ALPHAMELTS_FRACTIONATE_SOLIDS. To fractionate only certain phases instead, make sure this environment variable is unset and put 'Fractionate: phase' lines in your *melts_file*. By default, this will fractionate all exsolved instances of the given phase (e.g. plagioclase and K-spar, which are both treated as 'feldspar'; augite and pigeonite, which are both treated as 'clinopyroxene'). To fractionate just plagioclase, say, use the following procedure. First run alphaMELTS with feldspar fractionation turned off until two feldspars are present and from the screen or text output make a note whether plagioclase is the first or second feldspar printed. Then edit the *melts_file* to include the line 'Fractionate: feldspar 1', if plagioclase is first, or 'Fractionate: feldspar 2' if K-spar is first. If the chosen exsolved phase (plagioclase in this example) runs out and is dropped from the assemblage then fractionation of the phase group (i.e. feldspar) will be turned off completely. You can also adjust whether some or all instances of a phase are fractionated at the `alphamelts` menu prompt (option 8), which could be particularly useful if the exsolved phase tends to join the assemblage again later.

Enrichment file

The *enrichment_file* is used for source mixing, flux melting and assimilation calculations. It has the same format as the major and trace element composition lines in the *melts_file* (i.e. 'Initial Composition:' and 'Initial Trace:'). The trace elements must be in the same order as in the *melts_file*; otherwise the program will terminate with an error message. Major element entries are generally optional; the total mass of enriching agent may be set using an 'Initial Mass:' line or using menu options. See sections on menu option 12 ('Source mixer'), ALPHAMELTS_FLUX_MELTING and ALPHAMELTS_ASSIMILATE. There should be a 'Title:' line at the start.

Table file

The *table_file* is a space, comma or tab delimited file with rows containing bulk compositions, fO_2 and / or PT conditions. Together with a *melts_file*, which acts as a template, the *table_file* may be used to automatically generate multiple *melts_files*. `run_alphamelts.command` will make one new *melts_file* for each row of data in the *table_file*; the new *melts_files* will have the same name

plus 1, 2, 3, etc. before the '.' (or at the end if there is no filename extension). In the new *melts_file* the name of the *table_file* used to generate it plus the row number will be appended to the 'Title:' line in parentheses. Note that `run_alphamelts.command` should deal correctly with 'wrong' line-endings, so *table_files* exported from Excel can be used in alphaMELTS without further formatting, even on Mac OS X.

Any lines in the template *melts_file* that begin with 'Initial' or 'Log fO2' can (optionally) be replaced by a value taken from the table. All other lines will keep their original form and values that do not have 'Initial' or 'Log fO2' at the start of the line (e.g. 'Suppress:' lines) will have to be changed in the *settings_file* or at the `alphamelts` menu prompt. To be replaced, the column header must be exactly like the appropriate line in the *melts_file* but stripped of any 'Initial', 'Composition', 'Trace', 'Log fO2', white space, colon and numerical value (in other words there must be *melts_file* line something like 'Initial *columnname*: *value*', 'Initial Composition: *columnname value*' or 'Log fO2 *columnname*: *value*'). Case does not matter, nor does the order of the columns in the *table_file*; columns that do not match will be ignored. For example, 'Temperature' will match, but not 'T', and 'FeO' will match, but not 'FeO*'. Finally, 'Pressure' will match whereas 'P' runs the risk of being mistaken for phosphorus if that is included as a trace element.

`run_alphamelts.command` will keep reading in header lines from the *table_file* until at least one column title matches the *melts_file* in the sense given above. Columns can be deliberately ignored by renaming the header but it is important that ignored columns (in the used header line or amongst the values) do not contain any white space or commas within them e.g. 'Experiment' with values 'Label1', 'Label2' etc. is fine but 'Location' with values 'Sierra Nevada, CA', 'Big Island, HI' or 'T (deg C)' are not. Ignored header lines can contain white space and commas so, for example, the 'Excel-type' output from `column_pick.command` can generally be used as a *table_file*.

ALPHAMELTS_ASSIMILATE, ALPHAMELTS_FLUX_MELTING and Amoeba (menu option 17) can all be made to read in multiple *melts_files*, as generated using the '-t *table_file*' switch. When asked for the filename replace the index portion of the filename with one '?', as in 'file?.melts' for 'file1.melts', 'file2.melts', ... 'file25.melts'. If the *melts_files* have fixed width indices then put as many '?' as there are digits (e.g. 'file??.melts' for 'file01.melts', 'file02.melts', ...).

Trace Data file

The *trace_data_file* is optional (even when ALPHAMELTS_DO_TRACE is set) and has two main uses: (a) toggling between use of constant partition coefficients or calculation of variable partition coefficients for individual phase / element pairs and (b) specifying different partition coefficients if you do not wish to use the default values. The default (constant) partition coefficients in alphaMELTS are taken from *McKenzie and O'Nions* [1991; 1995]. Note that some elements (or phases) that are not included in that compilation may still appear in the example source compositions e.g. Mo in *Sun and McDonough* [1989]. Valencies and ionic radii for any calculations of $D(P,T,X)$ are taken from the references in Table 1 and citations therein. An example *trace_data_file* listing all the default values is provided ('default_trace_data.txt') but is intended as a reference only as these values are already included in the source code for the `alphamelts` executable. These parameters may be modified or values added for additional elements and / or phases by using 'Set D:', for constant D -values, and 'Valency:' and 'Ionic Radius:', for $D(P,T,X)$, as seen in the next example:

Set D: olivine Nb 0.005
Set D: feldspar Y 0.03
Constant D: garnet Sc
Variable D: clinopyroxene Sm
... see footnote⁶ regarding variable *D* in feldspar (plagioclase) ...
Ionic Radius: Eu 1.25
Valency: Eu 2

By default, constant partition coefficients are used for all trace element calculations but calculation of variable partition coefficients may be turned on for individual element / phase pairs with a ‘Variable D:’ line included in the *trace_data_file*. So, for example, to have variable partition coefficients calculated for REE in clinopyroxene but use constant (default) partition coefficients for everything else make and use a *trace_data_file* containing one ‘Variable D: clinopyroxene REE’ line for each element. As described below, setting ALPHAMEELTS_TRACE_DEFAULT_DPTX uses $D = D(P,T,X)$ for the all the phase / element pairs in Table 1, except those with ‘Constant D:’ lines in your *trace_data_file*. So, for example, to use variable partition coefficients for all the elements possible with clinopyroxene and garnet but constant partition coefficients for feldspar you could set the environment variable ALPHAMEELTS_TRACE_DEFAULT_DPTX to ‘true’ and then make and use a *trace_data_file* with one ‘Constant D: feldspar element’ for each element in the feldspar row of Table 1 (i.e. Li, Na, ...). Note that Eu is trivalent by default so there is no ‘Eu-anomaly’ with feldspar $D(P,T,X)$ unless you edit the *trace_data_file*, as shown (Eu²⁺ ionic radius from Shannon [1976]). We hope to add redox-dependent Eu partitioning in future.

If ALPHAMEELTS_DO_TRACE_H2O is set then mineral-melt partition coefficients for H₂O in olivine, orthopyroxene, clinopyroxene and garnet will be calculated using *Mosenfelder, et al.* [2005] and / or *Hirth and Kohlstedt* [1996] and will override any values set in the *trace_data_file* (see ‘[Trace elements and water partitioning](#)’). Partitioning of water in hydrous phases, including liquid, will satisfy water budget constraints. Mineral-melt partition coefficients for any other nominally anhydrous phases, e.g. feldspar, may be specified in the *trace_data_file* if desired. If ALPHAMEELTS_DO_TRACE_H2O is not set then H₂O partition coefficients from the *trace_data_file* will be used but then H₂O will not affect melting so this is not recommended (see the ‘[MELTS file](#)’ section). We appreciate that this set-up is not ideal and will be implementing more flexible partition coefficients in future versions of alphaMELTS.

Each type of line should be repeated, as required for each element (and phase as appropriate) that you wish to change / add data for. Elements / phases that are not listed in the file will retain their default parameter values (see ‘default_trace_data.dat’) and the choice of constant or variable partition coefficients from the environment. The order of the lines does not matter. If default partition coefficients are used then the *D*-values for elements used in isotopic ratios, such as Nd and Sr, are copied to each isotope before the *trace_data_file* is read. The same goes for the valency and ionic radius. This means that if you wish to change these parameters in the *trace_data_file* you must do so for each isotope you are using, whether you wish to give them all the same partition coefficients (e.g. ‘Nd143’, ‘Nd144’, ‘Sr87’, ‘Sr86’...) or different ones (as might be the case for ‘Li6’ and ‘Li7’).

⁶ The *Blundy and Wood* [1994] model is for plagioclase feldspar, whereas feldspar in alphaMELTS is a three-end-member solution of albite, anorthite and sanidine. As of version 1.7: if selected, the $D(P,T,X)$ option is only applied to feldspars where the molar proportion of sanidine is less than one third. More potassic feldspars default to the constant *D* value (which may also be user-defined). This allows different partition coefficients to be used multiple feldspar instances.

Note that you do not need a trace data file at all if you wish to use the values from *McKenzie and O'Nions* [1991; 1995] and either have variable partition coefficients turned off completely or have variable partition coefficients turned on for all of the combinations in Table 1. Similarly, if you add or change a few partition coefficients with 'Set D:' lines, all other phase / element combinations will keep their default partition coefficient values (i.e. from *McKenzie and O'Nions* [1991; 1995] or zero).

Settings file

The *settings_file*⁷ is the input file for environment variables and other starting conditions. If no name is given the script looks in the working folder for a *settings_file* with the default name of 'alphamelts_default_env.txt' (unless a different default path / file was specified during installation). If `run_alphamelts.command` is called with no command line arguments then the user will be asked to confirm that 'alphamelts_default_env.txt' is to be used as the *settings_file*. The example 'alphamelts_default_env.txt' in the *download_folder* lists default values of all the environment variables that are used to control the behavior of alphaMELTS (see 'Environment Variables'). If you wish to edit the provided 'alphamelts_default_env.txt' file, please make a copy in a different folder and edit the copy. If *settings_file* is not found, the script will stop. As well as the default *settings_file*, two other sample files ('frac_xtal_env.txt' and 'isentropic_melt_env.txt') are provided and are annotated to show how the *settings_file* may be modified to perform isobaric fractional crystallization and isentropic melting calculations respectively. The `run_alphamelts.command` script will write the settings used to call the `alphamelts` executable to a *log_file*, which is formatted so that it may be used as a *settings_file* in later runs.

```
ALPHAMELTS_VERSION pMELTS
ALPHAMELTS_MODE isobaric
ALPHAMELTS_DO_TRACE true
ALPHAMELTS_DELTAT -10
ALPHAMELTS_MINP 20000
    ... list all environment variable to be set ...
    ... continued from previous page ...
!ALPHAMELTS_CONTINUOUS_MELTING true
    ... comment out all environment variables not to be set with '!' ...
Initial Temperature: 1100.0
Log fO2 path: FMQ
Log fO2 delta: -1.0
    ... list melts_file style lines to be appended / replaced ...
!Initial Entropy: 260.0
    ... comment out melts_file style lines to be ignored ...
Suppress: none
Suppress: olivine
Subsolidus phases: olivine clinopyroxene orthopyroxene spinel
    ... subsolidus phases are used (optionally) in batch or automatic mode ...
```

⁷ This was previously known as the 'command_file' but was renamed to avoid confusion with the '.command' scripts. The '.command' file extension is required so that the scripts can be double-clicked on Mac OS X.

The format for most of the file is `ALPHAMELTS_ENVIRONMENT_VARIABLE variable_value`, as shown in the (isobaric batch melting) example above. Any line that starts with a ‘!’ will be ignored, which provides a quick way to switch environment variables on and off between runs. Initial values for parameters, such as P , T and fO_2 may be set here too as long as the format in the `melts_file` is either ‘Initial *parametername*: *value*’ or ‘Log fO_2 *parametername*: *value*’. The original `melts_file` will be copied to `melts_file_bak` and the lines from the `settings_file` will either replace the corresponding lines from the `melts_file` or be appended. If the `melts_file` contains, say, an ‘Initial Entropy:’ line that you wish to ignore then put ‘Initial Entropy: 0.0’ in the `settings_file`. ‘Suppress: *phase*’ lines will be appended to existing ‘Suppress:’ lines from the `melts_file` unless a ‘Suppress: none’ line is encountered; in the latter case all previous ‘Suppress:’ lines will be deleted. Further ‘Suppress: *phase*’ lines may be added after ‘Suppress: none’ to add phases back into the omitted list. When running in batch mode (see next section) a superliquidus start will be used by default but if the subsolidus phases are set then these will be used instead.

Batch file

The `batch_file`, if used, is simply a list of exactly what the user would type in if he were running the program by hand (see ‘[Runtime alphamelts menu options](#)’). The simple example below assumes that the initial P - T etc. have been set in the `melts_file` and that the mode is isentropic. Automatic mode generates a similar file, ‘`auto_batch.txt`’, for simple calculations (no adjusting of parameters) that can be renamed and modified for later use.

1	Option 1 to read file
my_melts_file.melts	Name of file
7	Option 7 to set reference entropy
260.0	Reference Entropy
4	Option 4 to execute until max / min P reached
0	Sub-solidus start
olivine	list of phases
spinel	(opx and cpx will be added automatically)
x	x to end list of phases
0	Quit

Output files

Main output files

Text file output is written automatically with one or more lines for each of the calculated equilibria along the thermodynamic path. The values are recorded at each step *after* equilibrium conditions are achieved and just *before* any fractionations (i.e. melt extraction or solid fractionation). A title, taken from the original `melts_file`, is printed first; six space-delimited files with names ending in ‘_tbl.txt’ are written. `System_main_tbl.txt` is the main output file and contains a summary of the calculation run, and system thermodynamic data; columns are explained in Table 2.

For the other output files, each line of data starts with pressure and temperature, followed by:

- `Bulk_comp_tbl.txt`: total mass and bulk composition of the system in wt % oxides.
- `Liquid_comp_tbl.txt`: liquid mass and compositions, in wt % oxides, or ‘---’ if sub-solidus.

- Phase_main_tbl.txt: for all solid and liquid phases present, the masses, selected thermodynamic data, (see Table 3), viscosity and Mg# (liquid only), molar formulas (except liquid), structure (pyroxenes and amphiboles only), and wt % compositions.
- Phase_mass_tbl.txt: the masses of all phases present in more compact format than the previous file (unless ALPHAMEELTS_QUICK_OUTPUT is set).
- Phase_vol_tbl.txt: the volumes of all phases present in more compact format than the Phase_main_tbl.txt file (unless ALPHAMEELTS_QUICK_OUTPUT is set).
- Solid_comp_tbl.txt: mass and bulk composition of the solid residue in wt % oxides.

Table 2.

Title	Meaning	Units
P	Pressure	Bars
T	Temperature	K (or °C [†])
Mass	Total mass of system	G
F	Melt fraction (by mass, relative to the current system mass, prior to any melt extraction)	
Phi	Melt fraction (as above, but by volume instead of mass)	
S	Total entropy of system	J.K ⁻¹
H	Total enthalpy of system	J
V	Total volume of system	cm ³
dVdP ×10 ⁶	Partial derivative of volume with respect to pressure	cm ³ .bar ⁻¹
dVdT ×10 ⁶	Partial derivative of volume with respect to temperature	cm ³ .K ⁻¹
Cp	Total heat capacity of system	J.K ⁻¹
logfO2- <i>buffer</i>	Oxygen fugacity, absolute values unless system is currently assigned to a buffer <i>and</i> was initially unbuffered [‡]	log 10 units
logfO2- <i>buffer</i>	Oxygen fugacity, relative to the initially assigned buffer, or absolute if initially unbuffered	log 10 units
rhol	Density of liquid	g.cm ⁻³
rhos	Density of bulk solid	g.cm ⁻³

[†] Normally temperature output is in Kelvin but if the ALPHAMEELTS_CELSIUS_OUTPUT variable is set then all file and screen output will be in °C. See the relevant environment variable entry for more details.

[‡] Where 'initially' refers to the system state recorded in the first row of the table. If system is currently buffered and was initially unbuffered, the value relative to the current buffer will be shown. Setting ALPHAMEELTS_SAVE_ALL and / or certain environment variables or menu options may cause the buffer to be switched on or off during execution. If ALPHAMEELTS_QUICK_OUTPUT is set then the system state from the first row of the table state is not saved and the two fO2(*buffer*) columns will contain the same values. Note that as of v1.8 *buffer* means e.g. "NNO", whereas previously it was "NNO+fO2Delta", so the tabulated values will now be fO2Delta, rather than 0.0.

Title	Meaning	Units
viscosity	log 10 viscosity of the liquid (Shaw model)	Poise
aH2O	Activity of water in the liquid	
chisqr	Chi-square fit to target composition(s) when running ‘Amoeba’	

If trace element calculations are being performed, an additional file ‘Trace_main_tbl.txt’ will be written. For each of the calculated equilibria along the path, this includes:

- Pressure, temperature, system mass and the bulk partition coefficient for each trace element.
- The mass and the composition in ppm or normalized units for each trace element (except H₂O, which is always in ppm) for the bulk system, solid residue, liquid and solid phases.

Table 3.

Title	Meaning	Units
Phase	Name of the phase	
Mass	Mass of the phase	g
S	Total entropy of the phase	J.K ⁻¹
H	Total enthalpy of the phase	J
V	Total volume of the phase	cm ³
Cp	Total heat capacity of phase	J.K ⁻¹

After `alphamelts` finishes, the output files are processed by `run_alphamelts.command`. ‘Phase_main_tbl.txt’ and ‘Trace_main_tbl.txt’ are tidied up and the entries sorted into separate tables, one for each phase etc.; a few other minor edits are aimed at making it easier to import the data in Excel or Matlab using `column_pick.command`. For convenience, the six or seven text files are concatenated into a single *output_file* as well; by default the name of the *output_file* is ‘alphaMELTS_tbl.txt’ but it may be set on the command line with ‘-o’ (see below).

Integrated output file

For the full integration (see ‘[Special calculations](#)’ and option 16 below) the first two lines of the *integrated_output_file* are the bulk properties of the melt regime if melting continues all the way to the base of the igneous crust (and no interpolation is performed). The crust is assumed to have a density of 2.62 g.cm⁻³. A table title is printed for use with `column_pick.command`. The first seven entries are described in Table 4.

Table 4.

Title	Meaning	Units
Pc	Thickness of the igneous crust for passive-flow (2-D) integration, in pressure units	bars
FB	Bulk or mean extent of melting (see <i>Plank, et al.</i> , [1995]), for 2-D integration with melting stopping at Pc (by mass, relative to starting mass, for extracted melt only)	

Title	Meaning	Units
Zc	Thickness of the igneous crust for 2-D integration	km
<P>base	Mean pressure of melt extraction	bars
Pc1D	Thickness of igneous crust for column-flow (1-D) integration, in pressure units	bars
Fmax1D	Maximum extent of melting reached at Pc1D (extracted melt only)	
Zc1D	Thickness of the igneous crust for 1-D integration	km

The remaining columns give the concentration of each oxide in wt % and each trace element, either in ppm or normalized to the chosen composition, in the 2-D aggregate primary liquid and then the same for the 1-D aggregate primary liquid.

The next two lines are the title and header for all lines to follow, and the remaining lines show aggregate properties at each depth (or increment of aggregate melt, if interpolated) in the melting regime. Integration begins at the solidus (Pressure = P_0) and stops at the depth of interest. The first six columns are described in Table 5. The remaining columns give the concentration of each oxide in wt % (or trace element, either in ppm or normalized to the chosen composition) in the partial aggregate primary melt, for 2-D integration from P_0 to P and then the same for 1-D integration from P_0 to P. If the pressure integral is not performed then the output will be P, T and F from Table 4 and the concentration of each oxide in wt % (or trace element, either in ppm or normalized to the chosen composition) in the partial aggregate primary melt up to each pressure (or temperature) point.

Table 5.

Title	Meaning	Units
P	Pressure	bars
T	Temperature	K (or °C)
F	Aggregated melt fraction at P (by mass, relative to starting mass, for 1-D integration of extracted melt only)	
P1D	Mean pressure for 1-D integration from P_0 to P	bars
P2D	Mean pressure for 2-D integration from P_0 to P	bars
iFdP	The integral of F with respect to pressure; this is the thickness of crust in pressure units generated between P_0 and P, for 2-D integration	bars

Thermodynamic phase output file

This file contains affinity and composition estimates for each metastable phase, similar to the dynamic screen output of the GUI version of MELTS, together with molar thermodynamic data for each phase in the current assemblage. The first line is a title, followed by a single line of bulk system information for the current conditions with a similar format to the one in Table 3. fO_2 is reported only relative to the current buffer, if any; the first value uses the chosen method of calculating fO_2 (e.g. *Kress and Carmichael* [1991]) and the second value always uses the thermodynamic method described in *Asimow and Ghiorso* [1998]. See the relevant environment variables in the ‘[P-T path and fO2 control](#)’ section for more details.

Next are the molar properties of oxygen gas, including the fO_2 , at the current P - T conditions. Most of the values are the same as the phase assemblage output that follows (see Table 6). The absolute fO_2 is reported in log units, together with the fO_2 relative to the FMQ buffer (regardless of the imposed buffer, if any). The first ‘logfO2-QFM’ value uses the equations in *Myers and Eugster* [1983] to calculate the FMQ buffer (i.e. no pressure term), whereas the second value uses a modified equation, consistent with Berman [1988] (see ALPHAMELTS_FO2_PRESSURE_TERM).

Output for each ‘stable’ phase (see menu option 15 for explanation of quotes) in the assemblage appears next and is as detailed in Table 6. For other (i.e. metastable) phases, these entries are replaced by the name and affinity of the phase, with respect to the current assemblage.

Table 6.

Title	Meaning	Units
Phase	Name of the phase	
moles	Number of moles of the phase	mol
S/mol	Molar entropy of the phase	J.K ⁻¹ .mol ⁻¹
H/mol	Molar enthalpy of the phase	J.mol ⁻¹
V/mol	Molar volume of the phase	cm ³ .mol ⁻¹
dVdP ×10 ⁶ /mol	Partial derivative of phase volume with respect to pressure	cm ³ .bar ⁻¹ .mol ⁻¹
dVdT ×10 ⁶ /mol	Partial derivative of phase volume with respect to temperature	cm ³ .K ⁻¹ .mol ⁻¹
Cp/mol	Molar heat capacity of phase	J.K ⁻¹ .mol ⁻¹

The remaining columns list for each end-member in turn: the formula, the molar proportion and, for stable phases only, the chemical potential of the pure phase (μ°) and the actual chemical potential of the phase (μ). For metastable phases, the end-member proportions are the corresponding composition estimate (see *Ghiorso* [1994]). Only those phases and end-members that fall within the bulk composition space will be printed; e.g., hornblende will not be printed in an anhydrous system.

Binary restart files

The *restart_file* contains, in binary format, all the information that the program needs to continue calculations from the current thermodynamic state. This can be useful if (pH) MELTS has trouble converging using the desired starting P , T conditions. See the ‘[What to do when \(pH\)MELTS misbehaves](#)’ section for more details. You can write a *restart_file* at any time using menu option 13. If the program has just executed then the restart file will be written for the final conditions *after* any fractionations, e.g melt extraction, solid fractionation. If you read in a *restart_file* at the start of a new `alphamelts` session (using option 11) the program state will be equivalent to that after reading in a *melts_file* and doing a single batch calculation (see menu option 3).

Menu options 11 and 13 provide, for *restart_files*, the same or similar functions that options 12 and 14 do for *melts_files*. Differences in the details reflect different ways in which *restart_files* and *melts_files* are likely to be used. Two specific examples of *restart_file* use are discussed briefly below and in more detail on the forum.

When reading in a *restart_file* it is always important that the environment variables are (almost) identical to those used when the *restart_file* was made. Otherwise you are likely to get segmentation

faults (e.g. because the program is / is not expecting trace element records in the *restart_file*). A little flexibility is allowed, so it is usually acceptable if environment variables from the ‘[P-T path and fO2 control](#)’ (includes ALPHAMEELTS_MODE) and ‘[Open vs. closed system](#)’ sections used when making the *restart_file* do not match the current settings.

Restart file use: isenthalpic assimilation

Another use for *restart_files* is isenthalpic assimilation calculations. In this mode, the program attempts to find a thermodynamic equilibrium state for the assimilant at a user-specified temperature. If you use a *melts_file* style input file for the assimilant then an equilibrium assemblage will be sought for the current system temperature, using a superliquidus starting solution, before extrapolating phase properties (i.e. enthalpy) to the desired assimilant temperature. The reason for this is that the subsolidus norm calculation is designed for peridotite / basalt compositions whereas many assimilants will be more granitic. As of alphaMELTS 1.3, you can provide one *melts_file* style input file for each mineral phase in the assimilant but this requires that microprobe data / mineral modes are available. Alternatively, you can use a fully equilibrated subsolidus start for the assimilant if you have previously saved one in a restart file. This can be useful if only the bulk compositions is known and the assimilant temperature and system temperature differ enough that the expected phase assemblages for that composition would not be the same at each temperature.

Restart file use: phase metastability

Finally, a *restart_file* may be used in the following workaround for problems with phase metastability. Consider an example where a composition of a tholeiite, with olivine phenocrysts and a groundmass including orthopyroxene, clinopyroxene, is used for a fractional crystallization run. Occasionally the routines for estimation of phase stability outlined in *Ghiorso* [1994; page 5495] and *Asimow & Ghiorso* [1998] fail and as a result one or more of the expected phases, in this case orthopyroxene, are not stabilized on cooling. Phase stability errors can occur even if the liquid composition is the result of an alphaMELTS partial melting calculation with, say, equilibrium liquid, olivine, orthopyroxene, clinopyroxene and spinel; one could also envision similar errors associated with using the composition of volcanic glass as a starting liquid. As the energetic differences between the competing pyroxenes are very small, the automatic workaround implemented in *Adiabat_1ph* 1.8 may not be sufficient and the problem can be difficult to solve by other means (see ‘[What to do when \(pH\)MELTS misbehaves](#)’). Instead, a binary *restart_file* can be used to add a ‘seed’ containing tiny amounts of liquid, olivine, two pyroxenes and spinel. The numerical problem is analogous to kinetic inhibition in a petrological experiment and the means of overcoming it is similar to adding a crystalline seed to an experimental starting material.

Command line switches

-h

If this switch is first, `run_alphamelts.command` prints the list of all available switches, with a brief explanation of each entry, and then either aborts or asks the user if they wish to start again (depending whether `run_alphamelts.command` is running in normal mode or double-click mode etc.). It also prints the version number of the script and which versions of the `alphamelts` executable it should be compatible with.

-f settings_file

Input file for environment variables and other starting conditions. If not specified the script will look for a file called 'alphamelts_default_env.txt'.

-b batch_file

Run `alphamelts` in batch mode. Menu options and filenames etc. that would be typed in during operation of `alphamelts` are instead passed from the user-supplied *batch_file* by the Perl script.

-a

Run `alphamelts` in automatic mode; this is similar to batch mode, except that the Perl script writes and then uses the simple batch file 'auto_batch.txt' for execution. It assumes the appropriate initial conditions have already been set in the *melts_file*, *settings_file* or *table_file* (see the 'Note about initial P, T, etc.'). It will use a subsolidus start with phases specified in the *settings_file*, if given, or a superliquidus start otherwise. When used with a *table_file*, automatic mode will run each new *melts_file* in turn.

-m melts_file

File for initial compositions, etc. If '-t' is used or if initial values are altered in the *settings_file* then `run_alphamelts.command` needs to know the name of the *melts_file* to edit before starting `alphamelts`. The *melts_file* will be copied to *melts_file_bak* before any changes are made. Specifying '-m' will ensure that the *melts_file* line-endings are correct; otherwise, '-m' is optional.

If you are running in automatic mode then the *melts_file* name will be passed to `alphamelts` by `run_alphamelts.command`. Otherwise, if you are not running in batch mode, the file name will be printed just before `alphamelts` is started so you can copy it when using the menu commands. When `alphamelts` is used in interactive mode (i.e. not automatic and not batch), specifying the '-m' option on the command line *does not* cause the input information contained in the *melts_file* to be read into `alphamelts`; you still need to use menu option 1.

As of alphaMELTS version 1.9: if a *melts_file* is specified in this way then lines in the file, such as 'Final Temperature:', may be used to set one or more of the following environment variables.

```
ALPHAMELTS_MAXT
ALPHAMELTS_DELTAT
ALPHAMELTS_MINT
ALPHAMELTS_DELTAT
ALPHAMELTS_MAXP
ALPHAMELTS_DELTAP
ALPHAMELTS_MINP
ALPHAMELTS_DELTAP
ALPHAMELTS_FRACTIONATE_SOLIDS
ALPHAMELTS_FRACTIONATE_WATER
ALPHAMELTS_CONTINUOUS_MELTING
ALPHAMELTS_FRACTIONATE_SECOND_LIQUID
ALPHAMELTS_MULTIPLE_LIQUIDS
ALPHAMELTS_MODE
```

This setup means that, for most simple calculations, the same *melts_file* can be used for alphaMELTS 1.9 and 2. For backwards compatibility between alphaMELTS 1.0 and versions 1.8.1 and earlier (which ignored lines like ‘Final Temperature:’) the ‘-m *melts_file*’ switch can be omitted. If that is not possible, e.g. because a *table_file* is being used, then there are two options: either remove any lines from the *melts_file* that may cause inconsistencies, or make sure to set the environment variable in the *settings_file*. For example, if the lines in the *melts_file* imply that ALPHAMELTS_DELTAT will have a value of -5 °C, then this can be overridden by putting ‘ALPHAMELTS_DELTAT -3’ in the *settings_file*. For more information about this transitional behavior, see the ‘[MELTS file](#)’ section.

-t *table_file*

This option can be used to automatically generate multiple *melts_files* from a single *melts_file* and a space-, comma- or tab-delimited *table_file* with rows of initial conditions (e.g. bulk composition) that are substituted into the original *melts_file* to generate new files. The name of the template *melts_file* must be passed to `run_alphamelts.command` with the ‘-m’ switch described above.

If you use automatic mode with a *table_file* then `alphamelts` will read in and execute each file in turn. It will use the same superliquidus / subsolidus settings from the *settings_file* for each *melts_file* (as described under ‘-a’ above). You would need to set ALPHAMELTS_SAVE_ALL to ‘true’ to get the output from the whole run. By editing the *batch_file* and / or with suitable settings you can tailor the multiple run to your needs. For example, by setting ALPHAMELTS_MAXP so that all initial *P-T* conditions are out-of-limits you can force `alphamelts` to run a single *P-T* point for a variety of bulk compositions.

-o *output_file*

`run_alphamelts.command` puts the main tables of output into separate files but also concatenates them into a single file. By default this file is called ‘alphaMELTS_tbl.txt’ but this switch, if set, can be used to give a different file name. If *output_file* ends in ‘_tbl.txt’ the file will be moved to the output folder defined using ‘-p’ (see next entry).

-p *output_path*

The path of the folder to put the output files in, either relative to the working folder or the absolute path. If the folder does not exist then it will be created. The main output files, any files set using environment variables (e.g. ALPHAMELTS_INTEGRATE_FILE) and all files ending in ‘_tbl.txt’ will be put in the *output_path* folder. When using the double-click mode of interaction it is more convenient to create the target folder manually and then drag-and-drop it to the command line, which avoids typing in a long pathname.

-c *column_file*

This switch automatically runs ‘`column_pick.command < column_file`’ when the text files have been processed by `run_alphamelts.command`. The *column_file* here is the command line input to the script (compare *melts_file_list* in the third `file_format.command` example of ‘[Interactive operation](#)’) and is not the same as the *column_list_file*. The *column_list_file* should appear in *column_file* with the *table_file* name (see ‘[The column_pick.command script](#)’). If *table_file* ends in ‘_tbl.txt’ the file will be moved to the output folder defined using ‘-p’, if any.

-l *log_file*

This file is a record of the environment variables, initial conditions and files used (including those set using ‘-m’ or ‘-o’). The contents of the *settings_file* are written to the *log_file* together with a message giving the time and date of the run. The *log_file* can be used as a *settings_file* for later runs. If ‘-l’ is not used this information will be written to ‘logfile.txt’.

Runtime alphamelts menu options

The menu commands have the following effects. The first four options will be used in almost all simple scenarios, whereas the next few are for slightly more complicated calculations. Options 11 – 16 are concerned with file input and output; the next two options are for specialized calculations and the last ones for menu display and quitting.

1. Read MELTS file to set composition of system

Asks for filename of the *melts_file*, opens it if possible, and sets bulk composition, starting pressure, temperature, log fO_2 path, reference entropy, enthalpy or volume, and which phases to suppress. Other information in the file is ignored. If you do not read in a *melts_file* with this option (or alternatively read in a *restart_file* with option 11) you will receive error messages if you try to use any other options until a file has been read in.

2. Twiddle starting or continuation parameters

Set or modify starting temperature (°C) and starting pressure (bars). Enter a value ≤ 0.0 to retain the old temperature and / or pressure.

If you have previously executed the program with option 4 and hit one of the maximum or minimum P or T limits then you will be asked if you would like to alter that limit and / or change ALPHAMELTS_DELTAT or ALPHAMELTS_DELTAP, as appropriate. This allows small changes in conditions (e.g. suppressing or allowing liquid or turning the fO_2 buffer on or off) to be made during a run. It also means that a superliquidus start can be used for *batch* melting of troublesome compositions (see subsolidus start in option 3). If you are in PT path mode, and have reached the end of the *ptpath_file*, you will be given an opportunity name a new *ptpath_file* instead of the maximum or minimum limits.

If the program does not second-guess the environment variables you need to alter or for more significant changes (e.g. having found a basaltic subsolidus assemblage, say, you wish to use it for *fractional* melting) then use option 13 to save a *restart_file*. Note that the order in which you call the options is important: if you read in a *melts_file* or *enrichment_file* at the menu or perform a single calculation either intentionally or inadvertently (i.e. the current P or T is still outside limits) then you may have to rerun alphAMELTS with new P - T limits or increments instead; this keeps the *batch_file* format simple and maximizes backwards compatibility.

3. Single (batch) calculation

This calculates the equilibrium state at the (current) starting conditions and then returns without putting any information in the output file buffer (unless ALPHAMELTS_SAVE_ALL is set). If *restart_file* input was used, the program assumes that at least one calculation was performed before creating the file and uses the assemblage therein as a start. Otherwise, the first time the user picks one of options 3 or 4 he is prompted for a superliquidus (1) or subsolidus (0) start. Subsequent calls use the assemblage of the previous calculated equilibrium until a new *melts_file* or *enrichment_file* is read in at the menu, in which case the question may be reposed.

On subsolidus start the user is asked for the phases to use. The norm routine is biased towards peridotites and, to a lesser extent, basalts (*sensu lato*) so the list must include clinopyroxene and orthopyroxene (even if one pyroxene is later dropped from the assemblage), plus one of olivine or quartz. If bulk composition includes Cr₂O₃ then spinel is needed and K₂O requires feldspar. Type 'x' when you are done with the list. Any fO_2 buffer will be switched off temporarily so that the equilibrium assemblage can be refined with a single run through the minimization routine. Then the full fO_2 -buffered calculation is attempted.

For single calculations no liquids or solids are removed (regardless of whether or not ALPHAMELTS_CONTINUOUS_MELTING or ALPHAMELTS_FRACTIONATE_SOLIDS are set). Fractionation of liquid or solids will start on execution. Similarly, for isentropic, isenthalpic or isochoric calculations the reference conditions will not be set until option 4 is chosen, unless you set them manually (see option 7).

If phase diagram mode is on the program will then ask for a phase name and find the limit of appearance of that phase or, for liquid, the point where the melt fraction or a_{H_2O} has a particular value. Usually the boundary will be located by bisection in temperature at the given pressure; for solid phases, if ALPHAMELTS_DELTAP is set to zero and ALPHAMELTS_DELTAT is non-zero, temperature will be fixed and pressure varied. In order to mimic the 'Find Liquidus' menu option in the GUI version of MELTS, set phase diagram mode to 2 and choose 'liquid'. Remember to turn phase diagram mode off before calling menu option 4 (or choose 'x' to return to the menu and try again). See menu option 10 for more details.

4. Execute (follow path, mineral isograd or melt contour)

Begins normal execution at initial conditions, like option 3, and continues along the chosen P - T path or, if phase diagram mode is on (see option 10), along a chosen boundary or contour (see option 3). Pressure or temperature is incremented and equilibria are calculated at each point until the program fails or a limit (maximum or minimum pressure or temperature, end of *ptpath_file* or maximum number of iterations) is reached.

If no calculations have been performed previously, the user is prompted for a superliquidus (1) or subsolidus (0) start; see instructions for option 3 for details. If the mode is isentropic, isenthalpic or isochoric and the reference entropy, enthalpy or volume have not been set manually then one calculation will be performed at the current P and T and the resulting thermodynamic state used to set the relevant reference property. Any H₂O buffer and any fO_2 buffer will be switched off at this point also, unless ALPHAMELTS_IMPOSE_FO2 is set.

If phase diagram mode is on the algorithm will usually search for a stable reaction involving the chosen phase. If phase diagram mode is set to 2, it will track the affinity=0 contour. For the purpose of calculating phase diagrams, it should not matter what ALPHAMELTS_MODE is set to. For most reactions it is fine to specify the pressure increment with ALPHAMELTS_DELTAP. For subsolidus reactions with particularly steep Clapeyron slopes the routine may work better if you set a temperature increment with ALPHAMELTS_DELTAT and set ALPHAMELTS_DELTAP to zero.

5. Set fO_2 buffer

A menu of possible fO_2 buffers is printed. If a value between 1 and 4, inclusive, is input then the user will be prompted for an offset in fO_2 log units (see 'Melts File' section). A value of 0 switches off fO_2 buffering; any other value leaves the fO_2 buffer and offset unchanged. For

isentropic, isenthalpic or isochoric modes, the fO_2 buffer may be imposed for single (option 3) calculations but will switch off automatically once the reference entropy, enthalpy or volume has been set in option 7 or option 4 (unless you set ALPHAMEELTS_IMPOSE_FO2; see the discussion in ‘[Open versus closed system behavior](#)’).

6. Set H2O (ppm) or aH2O

If water is being treated as a trace element (i.e. if ALPHAMEELTS_DO_TRACE_H2O is set) this gives the user an opportunity to reset the current H₂O content in ppm without needing to read in an *enrichment_file*. Otherwise, the user will be asked to set aH_2O . If a value $0 < aH_2O \leq 1$ is entered then water buffering will be switched on. This will adjust the wt % water content set in the *melts_file* to maintain the required activity in the melt. If aH_2O is set to 0 then H₂O buffering will not be switched on (or will be switched off if previously used). Like the fO_2 buffer, the H₂O buffer switches off automatically once the reference entropy, enthalpy or volume is set in isentropic, isenthalpic or isochoric modes. (There is currently no ALPHAMEELTS_IMPOSE_FO2 equivalent, although should user-demand arise it could be introduced; see ‘[Open versus closed system behavior](#)’).

7. Impose initial entropy, enthalpy or volume

The program checks whether you are running in isentropic, isenthalpic or isochoric mode and asks for the total extensive reference entropy (in J.K⁻¹), enthalpy (in J) or volume (in cm³). Entering ‘0’ will temporarily turn off the constraint, if there is one, or leave it unset. If any calculations have been performed then the current total entropy, enthalpy or volume can also be selected. The temperature, or pressure in the case of isochoric mode, will be adjusted accordingly just before the next calculation. If this option is not used, or if ‘0’ is selected, then the relevant reference condition will be calculated for the initial P - T conditions when option 4 is called.

8. Adjust solid phase setting(s)

Set whether a particular phase is allowed or suppressed, overriding any earlier setting from the *melts_file*. If the phase is allowed then the user will be given the option to adjust the selective fractionation settings of the phase. This may be useful if you are fractionating just one instance of a particular phase and it rejoins the assemblage after previously being exhausted (see ‘[MELTS file](#)’). It also allows for turning off fractionation of individual phases when the environment variable ALPHAMEELTS_FRACTIONATE_SOLIDS has been set.

9. Turn liquid on / off

Set whether liquid is allowed or suppressed. This option can be useful if you want to perform calculations at a particular potential temperature (T_p). Initially suppress liquid to find the liquid-absent path and then switch it on for melting calculations (use option 2 to reverse the sense of ALPHAMEELTS_DELTAP, if appropriate).

10. Turn phase diagram mode on / off

When turned on, phase diagram mode alters the behavior of options 3 and 4 so that (once at least one equilibration calculation has been performed) a routine is started that seeks the boundary where a phase appears / disappears from the assemblage or where the melt fraction or aH_2O have particular values. This option allows phase diagram mode to be turned on or off; see options 3 and 4 for more details on operating in phase diagram mode.

If phase diagram mode is set to '1', then for solid phases you must specify on which side of the desired reaction the phase is stable (e.g. choose 'high pressure' for the low pressure limit of garnet stability). In the case of feldspar or clinopyroxene reactions, the user must also tell the program how many copies of the phase to consider. Normally the answer will be '1' but for a K-spar boundary in the presence of plagioclase it would be '2', as these are both treated as 'feldspar'; likewise pigeonite and augite are both 'clinopyroxene' as far as MELTS is concerned.

For liquid in phase diagram mode '1', you must state what melt fraction or water activity to look for. Then choose whether to track the water activity or melt fraction and, if the latter, whether it should be measured by mass (F) or volume (Phi). Tracking a line of constant a_{H_2O} should only be tried if ALPHAMELTS_DO_TRACE_H2O is unset and the water buffer is off (see option 5). Constant a_{H_2O} contours may only be tracked in the region $0.0 < F < 1.0$ and if a boundary is intersected then the solidus or liquidus will be tracked temporarily instead.

A slower but more rigorous search algorithm can be used for phases that exhibit metastability (e.g. if hysteresis in the melt fraction is observed). Nevertheless, sometimes the algorithm has trouble finding the exact solidus (e.g. see '[What to do if \(pH\)MELTS misbehaves](#)') and so, if doing continuous melting, it may be more successful to use the *effective solidus* instead i.e. the melt fraction above which melt is extracted. This value, equal to either ALPHAMELTS_MINF or ALPHAMELTS_MINPHI, is the default if it is set and may be chosen by giving a negative value for the desired melt fraction. Otherwise the default is 0.0.

Normally phase diagram mode will search for a stable reaction involving the chosen phase and any other phases that are not actively suppressed. If phase diagram mode is set to '2', then for solids the boundary where the phase is saturated in the bulk liquid composition will be sought instead, whilst ignoring the potential crystallization of other phases. See *Asimow and Longhi, [2004]* for a comparison of stable versus metastable liquidus diagrams. On execution, the user is asked which solid phase to search for and whether the phase is present on the high- or low-temperature side of the boundary.

For liquid in phase diagram mode '2', the algorithm will search for the point where any solid phase except water is just saturated. This is similar to the 'Find wet liquidus' function in MELTS_Excel. Note that trace element calculations are only available for phase diagram mode '1' but diagram mode calculations are significantly quicker. Most phase diagram mode '1' calculations will stop where the phase of interest (or liquidus phase) is just stable so its chemistry can be seen. Phase diagram mode '2' tends to stop where the phase of interest is just out.

Phase diagram mode behavior used to be set with ALPHAMELTS_METASTABLE_ISOGRAD. This environment variable is now deprecated, though will work as before for the time being.

11. Update state using restart file

If no *melts_file* or *restart_file* has been read in so far, this option reads in the system thermodynamic state from a *restart_file*. If you are using this option to continue calculations after a program crash, it is *crucial* that you do this first (i.e. do not read in a *melts_file* before choosing option 11) and that you use almost exactly the same set of environment variables etc. as was used when the *restart_file* was generated; check the '[Binary restart files](#)' section for a few environment variable changes that can be tolerated. If you do not read in a *restart_file* (or alternatively read in a *melts_file* with option 1 or 12) you will receive error messages if you try to use any other options until a file has been read in.

You are given the option to reset the total mass and reference mass values, which is equivalent to editing the ‘Initial Mass:’ line before reading in a *melts_file*. The reference mass is used to scale certain thresholds (e.g. the limit below which a component is considered to be exhausted) and is the ‘starting mass’ used in melt extraction (see the ‘[Integrated output file](#)’ section). If you are restarting a continuous melting calculation you should leave the masses untouched, but otherwise you will probably want to reset the reference mass to the total mass, or reset both to a new value. The composition and masses of all phases in the system will be scaled accordingly.

If a *melts_file* or *restart_file* has already been read in, then this option will call ‘Source Adder’, which is intended for adding material that is *close to equilibrium* with the current bulk composition. The user is given the option to (1) keep the current state, but scale the masses as described above, and return or (2) read in a *restart_file*, scale the masses of the new assemblage and add it to the current assemblage on a phase by phase basis or (3) read in a *restart_file*, scale the masses of the new assemblage and use it to replace the current state. Thus, two calls to ‘Source Adder’ may be required to sum two assemblages in the desired quantities. The third option is equivalent to calling option 11 when no *melts_file* or *restart_file* has been read in.

Note: if no calculations have been performed since reading in the original *melts_file* then the current assemblage will be treated as all liquid. On the next call to option 3 or 4, the resulting phase assemblage will be used as the starting guess. Adding trace elements is optional. See the ‘[Restart file use: phase metastability](#)’ section for an outline of the intended use of this combination of menu option and file type, and the forum for a more detailed explanation.

12. Update composition using MELTS file

If no *melts_file* or *restart_file* has been read in so far, this option reads in a *melts_file* and sets up the system (and is, in fact, the same as using option 1). More normally, this option is used to call the ‘Source Mixer’ function. Use it for one-off calculation of a new source composition e.g. to model melting of homogenous mixtures of differing proportions of peridotite and pyroxenite.

The program will ask for an *enrichment_file* name. A new bulk composition is calculated with given proportions for major and trace elements or trace elements only, as indicated by the user. If major elements are included then the assemblage will be reset and any reference quantity that has been specified (e.g. entropy, enthalpy or volume; see option 7 below) will be unset. On the next call to option 3 or 4 the user will be asked to choose between a superliquidus or subsolidus starting-guess, regardless of whether any calculations have been performed previously.

13. Write out restart file

Usually used after at least one equilibration calculation has been performed, this option allows you to save (in binary format) all the information the program needs to start calculations at the current set of conditions. If option 4 has just been run, this will be the state after any fractionation of solids or liquids but before, say, the addition of another increment of assimulant or fluid.

14. Write out MELTS file

Write a *melts_file* style input file that can be used for future calculations after minimal manual editing. The user is asked for a file name and given the choice to write the current *melt* composition or the current *residue / bulk* composition, if appropriate (i.e. not below the solidus, above the liquidus or doing perfect fractional melting calculations). If solid phases, as opposed to just water, are being fractionated the *bulk* composition alternative is offered. Otherwise the

residue composition can be written here and it is generally more useful to use option 13 to write a binary restart file for the *bulk* composition.

15. Write thermodynamic output for all phases

Asks for filename, opens it if possible, and writes thermodynamic output information for individual phases, as described in the ‘[Thermodynamic phase output file](#)’ section. If the name of the thermodynamic output file ends in ‘_tbl.txt’ then it will be moved to the output folder defined using the ‘-p’ command line switch, if any.

If this option is called just after a *melts_file* is read in, but before any equilibration, then writing out the thermodynamic output is counted as a ‘calculation’ for ALPHAMELTS_SAVE_ALL. By default, all solid phases will be metastable with respect to the bulk liquid composition. An option is given to ‘assimilate’, but not equilibrate, solid phases so that their thermodynamic properties can be output as well (for more details see the option for providing separate *enrichment_files* for each mineral phase with ALPHAMELTS_ASSIMILATE). The assimilated mineral will be treated as if it were a stable assemblage phase and its actual composition printed in Phase_main_tbl.txt if ALPHAMELTS_SAVE_ALL is set; the onus is on the user to provide mineral compositions that are close to stoichiometric in the (p)MELTS system.

Even if no equilibration is performed, the phase compositions will be updated to reflect the current fO_2 buffer. Without an fO_2 buffer, this routine can function as a generalized oxybarometer. For subsolidus fO_2 calculation make sure the original ‘liquid’ *melts_file*, i.e. the one entered with menu option 1, has 0.0 grams for all the major element oxides. This is the only situation for which it is valid to use a *melts_file* with zero total mass. The calculated fO_2 estimate will be more reliable if analysed compositions for multiple mineral phases are input; please remember that it is just an estimate (see the ‘[Disclaimer](#)’ about applying the results in an experimental setting).

16. Calculate integrated melt and output file(s)

In addition to standard output file(s), writes a table of aggregate crustal thicknesses and liquid compositions for 1-D and 2-D melting regimes. This assumes that melting is fractional or continuous and that pressure was being decremented. Generally, the user must choose between the full pressure-integral or a simple weighted sum of the extracted melts, and is then given options to use a simple rectangle rule or Cubic Gaussian Quadrature for integration. The former is usually adequate, unless the pressure (or melt fraction) points are unevenly spaced. If you choose the latter, or if the ALPHAMELTS_INTEGRATE_PHI variable is set, you will be given the option to output interpolated values in equal spaced increments of melt fraction (by mass or volume, as appropriate). You can interpolate the melt fraction: either (a) relative to the system mass / volume, corresponding to equal increments in aggregate extracted melt, or (b) relative to the starting mass / volume, which approximates the extraction events being triggered at a particular value of the instantaneous melt fraction (equivalent to F / Phi in Table 2, respectively).

The major element and, if ALPHAMELTS_DO_TRACE is set, trace element output file names must also be input. Afterwards, there is an option to write basic *melts_file* style input files (‘Initial Composition:’ and ‘Initial Trace:’ lines only) for the 1-D and 2-D average crust compositions or the last aggregate melt composition(s) if the integration did not reach the base of the crust or the pressure integral was not performed (e.g. for isobaric melting).

It is possible to use this menu option before any melt is extracted so that the *integrated_output_file* may be generated after a program crash. An intermediate file is specified

by the ALPHAMEELTS_INTEGRATE_FILE environment variable. In the first run, i.e. the melting calculations, this is an output file; when alphaMELTS is restarted, the same file automatically becomes the input file (as long as option 16 is called before option 4).

17. Fit parental melt composition (amoeba)

This option will start the 'Amoeba' routine to find the best-fit parental melt that can generate a given target composition or liquid line of descent by forward fractionation. The environment variable ALPHAMEELTS_FRACTIONATE_TARGET must be set for this to work correctly. When option 17 is called the current bulk liquid composition (e.g. a *melts_file* composition that could be an analysed melt inclusion, or generated by a previous backwards fractionation or melting calculation) will be used for the starting solution. The value of ALPHAMEELTS_MGO_TARGET (or MGNUMBER_TARGET) will be used to determine the MgO content of the primary liquid; the wt % of other oxides will be adjusted to reflect the new MgO wt % and to impose any fO_2 buffer.

The program will ask for one or more *melts_file*, which give the target liquid composition. To get the adjusted starting solution, choose '0' for the number of target *melts_files*. To use multiple *melts_files*, see the 'Table file' section for the filename format. Trace elements can be input and the primary liquid composition will be adjusted accordingly but the trace elements do not influence the amoeba minimisation. Amoeba performs many forward fractionation calculations, during which the screen and file output are suppressed, even if ALPHAMEELTS_SAVE_ALL is set. When it is finished, the current state will be the best estimate of the primary liquid. You can use option 3 to record it, if ALPHAMEELTS_SAVE_ALL is set, or option 4 to redo the 'best' forward calculation with full output, stopping when the MgO matches that of the most evolved target liquid. The 'chisqr' value (see Table 2) will be for the most recent amoeba fit and will not be reset unless, for example, a new *melts_file* is read in or menu option 17 is called again.

The 'standard errors' file may, optionally, be used to choose which oxides to vary, which to include in the chi-square goodness of fit estimate and to adjust the weightings given to each oxide. The program expects a *melts_file* format with absolute values, in wt %; use a zero to fix a particular oxide, or a negative number to fix it but still include it in chi-square. The default values used are given in the example file 'Amoeba.melts'. MgO is never included in chi-square as its value is used to determine where to stop the forward fractionation runs. SiO₂ will be calculated by difference from 100 wt %, so is not varied as such but may be included in chi-square. By default chi-square is calculated with all Fe as FeO* and, if there is no fO_2 buffer, Fe₂O₃ will be varied if FeO is being varied. A non-zero standard error for Fe₂O₃ will mean FeO and Fe₂O₃ are treated separately. If the fO_2 buffer is set then it is FeO* that is varied. At the start of each forward fractionation trial the buffer is imposed to get initial FeO and Fe₂O₃. The buffer is then turned off temporarily during the forward fractionation calculation, allowing amoeba to explore the composition space. H₂O may be varied but it may be more instructive to fix its content *a priori* and then rerun the amoeba routine for different choices. We have not been able to get amoeba to run when *aH₂O* is set, but setting ALPHAMEELTS_FRACTIONATE_WATER can be used to simulate water-saturated conditions instead.

18. Cumulate Invertor (not yet implemented)

This option is a placeholder for an as yet unfinished feature.

-1. Turn off (or on) menu display for options 1-18

Toggles menu display to reduce screen output (menu options still work if not displayed).

0. Quit

Exits. State information may be lost.

Note about initial P, T, etc.

Note that P , T and fO_2 may be initialized in the *melts_file* or the *settings_file*. P and T may also be ‘twiddled’ within the program with menu option 2 and fO_2 fixed with option 5. Reference Entropy, Enthalpy or Volume may also be input with an ‘Initial reference_value:’ line in the *melts_file* / *settings_file* / *table_file* or set with option 7. However, if initial values are given in a *table_file* they will overwrite *settings_file* values, which will in turn overwrite *melts_file* values. Similarly, menu options will override file input. Also, the P , T values that are recorded in the *log_file* (if any) will be those from the *settings_file*, even if they are reset later.

Environment Variables

Environment variables read from the *settings_file* are set by the `run_alphamelts.command` script before `alphamelts` is started. A default *settings_file* ‘`alphamelts_default_env.txt`’ is provided with a list of all environment variables and their default values (that is, the values they would be assigned at runtime if an empty *settings_file* were used). Note that within `alphamelts` environment variables are treated as ‘false’ if they have no value and ‘true’ if they have been set to any value (even the string value ‘false’!). In the sample file ‘`alphamelts_default_env.txt`’, some lines are commented out with ‘!’ as their default state is to be unset. The following environment variables affect alphaMELTS execution. Note that they are case sensitive.

Solution models

ALPHAMELTS_VERSION

Values: ‘`MELTS`’, ‘`pMELTS`’

Default: ‘`pMELTS`’

Set to choose the liquid thermodynamic model. As discussed above, `pMELTS` is only recommended for peridotite bulk compositions between 1 and 4 GPa. If doing `pHMELTS` calculations then use ‘`pMELTS`’ for melting and ‘`MELTS`’ for low-pressure crystallization.

ALPHAMELTS_OLD_GARNET

Values: *any*

Default: *unset*

Uses the old, incorrect, garnet model. When set, the behavior will resemble that of all GUI versions of MELTS released before 2005 (including Java MELTS) and the default behavior of `Adiabat_1ph` versions 1.4 – 2.X. When unset, will behave as if `ADIABAT_FIX_GARNET` was set in older (pre-3.0) versions of `Adiabat_1ph`.

ALPHAMELTS_OLD_SPINEL

Values: *any*

Default: unset

Use the spinel model from *Sack and Ghiorso* [1991], with perfectly ideal mixing of volumes. When set, gives the same behavior as Java MELTS and earlier (pre-3.0) versions of *Adiabat_1ph*. When unset, uses the spinel volume model as formulated in *Ghiorso and Sack* [1991].

ALPHAMEELTS_OLD_BIOTITE

Values: any

Default: unset

Uses the biotite model from *Sack and Ghiorso* [1989] instead of the default alphaMELTS one. When set, the program uses the same code as GUI versions of MELTS (2005-present). The behavior may differ slightly from earlier GUI versions of MELTS as a small error was fixed.

ALPHAMEELTS_2_AMPH

Values: any

Default: unset

Uses separate orthoamphibole and clinoamphibole phases. When set, gives the same behavior as Java MELTS and Corba MELTS applets, and earlier versions of *Adiabat_1ph* (which inherited its code from the Java MELTS branch). When unset, will use a single cummingtonite-grunerite-tremolite solution and deduce the amphibole structure from the energetics. See '[Alternative garnet, spinel, amphibole and biotite models](#)' for more details.

ALPHAMEELTS_NO_CHLORITE

Values: any

Default: unset

As of version 1.6, the chlorite model of *Hunziker* [2003] is included in alphaMELTS (see *Smith et al.* [2007]). To reproduce previous results for bulk compositions that may stabilize chlorite, add a 'Suppress: chlorite' line to the *melts_file* or set this variable.

P-T path and fO2 control

ALPHAMEELTS_MODE

Values: 'geothermal', 'isenthalpic', 'isentropic', 'isobaric', 'isochoric', 'isothermal', 'ptfile' (formerly 'ptpath'), 'ptgrid', 'tpgrid'

Default: 'isentropic'

Sets the calculation mode. This variable is case insensitive. 'PTfile' ('PTPath' is also accepted for backwards compatibility) reads in *P* and *T* from a file; see below. 'PTgrid' and 'TPgrid' perform calculations on a grid bounded by the starting *T* and ALPHAMEELTS_MINT/MAXT, and by the starting *P* and ALPHAMEELTS_MINP/MAXP. For 'PTgrid' the *T*-loop is inside the *P*-loop, whereas 'TPgrid' is the other way round; in either case the (superliquidus or subsolidus) starting solution migrates with the outer loop. Generally, unless liquid is suppressed, 'PTgrid' with ALPHAMEELTS_DELTAT < 0 will be the easiest combination. For other modes, there are various ways to initialize *P*, *T* and / or reference entropy, enthalpy or volume, and the

thermodynamic path is set using `ALPHAMEELTS_DELTAP` and / or `ALPHAMEELTS_DELTAT`. The settings are described elsewhere in this documentation.

ALPHAMEELTS_PTPATH_FILE

Values: filename

Default: unset

Gives the name of the *ptpath_file*, which is a simple space delimited text file with one ‘*P_value T_value*’ pair per line. If `ALPHAMEELTS_DELTAP` and `ALPHAMEELTS_DELTAT` are both zero the user will be asked for a maximum number of iterations to perform.

ALPHAMEELTS_DELTAP

Values: integer or float in bars

Default: +1000

This sets the pressure increment for isentropic, isothermal, geothermal or phase diagram mode. This is a signed number; i.e., a positive value steps upwards in *P*, negative steps down. If using a *ptpath_file*, a non-zero `ALPHAMEELTS_DELTAP` means the whole will be read in and executed. Setting `ALPHAMEELTS_DELTAP` to zero, with a non-zero value for `ALPHAMEELTS_DELTAT`, means that phase diagram mode will step in temperature instead of pressure (except for liquid).

ALPHAMEELTS_DELTAT

Values: integer or float in °C

Default: +10

This sets the temperature increment for isobaric, isochoric, geothermal or phase diagram mode. This is a signed number; i.e., a positive value steps upwards in *T*, negative steps down. If using a *ptpath_file*, a non-zero `ALPHAMEELTS_DELTAT` means the whole will be read in and executed. For phase diagram mode to step in temperature `ALPHAMEELTS_DELTAP` must be set to zero.

ALPHAMEELTS_MAXP

Values: integer or float in bars

Default: +30000 if ALPHAMEELTS_VERSION = MELTS; +40000 otherwise

Sets the maximum pressure the program will go to on execution.

ALPHAMEELTS_MINP

Values: integer or float in bars

Default: +1 if ALPHAMEELTS_VERSION = MELTS; +10000 otherwise

Sets the minimum pressure the program will go to on execution.

ALPHAMEELTS_MAXT

Values: integer or float in °C

Default: +2000

Sets the maximum temperature the program will go to on execution.

ALPHAMEELTS_MINT

Values: integer or float in °C

Default: 0

Sets the minimum temperature the program will go to on execution.

ALPHAMEELTS_ALTERNATIVE_FO2

Values: any

Overrides: ALPHAMEELTS_LIQUID_FO2

Default: unset

Normally, the parameterization of *Kress and Carmichael* [1991] is used to calculate fO_2 in the liquid. If conditions are subsolidus or liquid is suppressed then the approach detailed in *Asimow and Ghiorso* [1998] is used to construct an appropriate redox reaction to solve for fO_2 of the bulk assemblage. If this environment variable is set, however, then the method of *Asimow and Ghiorso* [1998] is used to calculate fO_2 regardless of whether liquid is present and so, in theory, allows for a smoother transition across the solidus.

ALPHAMEELTS_LIQUID_FO2

Values: any

Default: unset

Overridden by: ALPHAMEELTS_ALTERNATIVE_FO2

The method of *Asimow and Ghiorso* [1998; see above] is computationally more involved than the parameterization of *Kress and Carmichael* [1991] and it is not uncommon for fO_2 calculations to be successful with liquid present but fail subsolidus. It is possible to turn off the fO_2 buffer manually (option 5). Alternatively, if this environment variable is set then the fO_2 buffer, as formulated in *Kress and Carmichael* [1991], will only be imposed when liquid is present. Note that setting this variable does not change the fO_2 buffer setting (e.g. 'FMQ'); the program just ignores the flag if no liquid is around.

ALPHAMEELTS_IMPOSE_FO2

Values: any

Default: unset

Normally, for isentropic, isenthalpic and isochoric modes any fO_2 buffer will be switched off on execution once the reference entropy, enthalpy or volume has been set (usually after the first calculation or before if set manually). If this environment variable is set then the program will alternate between (1) an unbuffered isenthalpic / isentropic / isochoric step and (2) an isobaric / isothermal fO_2 buffered step. Overall this approximates an isenthalpic, isentropic or isochoric path with a desired fO_2 buffer (see '[Open versus closed system behavior](#)' and menu option 5).

ALPHAMEELTS_FO2_PRESSURE_TERM

Values: any

Default: unset

In most versions of MELTS, including Corba MELTS and alphaMELTS, by default, reference fO_2 buffers in the system Fe-Si-O are calculated from the equations given in *Myers and Eugster*

[1983]. In the standalone GUI version of MELTS a pressure term is added to the fayalite-magnetite-quartz buffer (FMQ) to give:

$$\log(fO_2) = -24441.9/T + 0.110(P - 1)/T + 8.290$$

When this environment variable is set then alphaMELTS uses the equation as shown whereas by default the second term is omitted. Inclusion of the pressure term can have subtle effects on the stability of certain phases such as pyroxenes. Note that in *Kessel et al.* [2001] the coefficient on the pressure term is 0.05, rather than 0.110; the value of 0.110 was chosen by Ghiorso to maintain consistency with Berman [1988].

Open vs. closed system

ALPHAMEELTS_CONTINUOUS_MELTING

Values: any

Default: unset

By default, batch melting equations are used. Setting this environment variable will change the melting mode to continuous or fractional, where melt is extracted after each equilibrium. Set ALPHAMEELTS_MINF to 0 for perfect fractional melting. In practice though, the program will run more smoothly if ALPHAMEELTS_MINF is slightly greater than 0. See the next four environment variables' entries for more details.

ALPHAMEELTS_MINF

Values: $0 \leq \text{float} < 1$

Default: 0.005

Overridden by: ALPHAMEELTS_MINPHI

If ALPHAMEELTS_CONTINUOUS_MELTING is set, then by default a fixed melt fraction, by *mass*, marks the threshold above which melt is extracted. This variable is used to change the amount of melt retained. If the current melt fraction (F in Table 2) is less than ALPHAMEELTS_MINF then all the melt will be retained until the next step, otherwise the amount of melt removed (approximately F - ALPHAMEELTS_MINF) will be adjusted so such that the melt fraction is exactly ALPHAMEELTS_MINF after extraction.

ALPHAMEELTS_MINPHI

Values: $0 \leq \text{float} < 1$

Overrides: ALPHAMEELTS_MINF

Default: unset

Overridden by: ALPHAMEELTS_CONTINUOUS_RATIO

If ALPHAMEELTS_CONTINUOUS_MELTING is set, then set this environment variable controls the retained melt fraction, by *volume* i.e. the 'residual porosity'. If the current melt fraction (phi in Table 2) is less than ALPHAMEELTS_MINPHI then all the melt will be retained until the next step, otherwise the amount of melt removed (approximately phi - ALPHAMEELTS_MINPHI) will be adjusted so that the melt fraction, by volume, is exactly ALPHAMEELTS_MINPHI after extraction.

ALPHAMEELTS_CONTINUOUS_RATIO

Values: $0 \leq \text{float} < 1$

Overrides: ALPHAMEELTS_MINPHI

Default: unset

Overridden by: ALPHAMEELTS_CONTINUOUS_VOLUME

This implements another alternative definition of continuous melting. Instead of extracting all liquid above a fixed mass or volume fraction, this option, if set, causes the program to multiply the liquid mass by a fixed ratio.

ALPHAMEELTS_CONTINUOUS_VOLUME

Values: any

Overrides: ALPHAMEELTS_CONTINUOUS_RATIO

Default: unset

This option, if set, extracts the required amount of melt to retain a constant total volume. This reference volume is set the first time melting occurs and is equal to the solid volume plus whatever melt volume is retained according to the ALPHAMEELTS_MINPHI variable (or a default value of 0.002 if that is unset). Only for isobaric or isothermal calculations.

Note that this is not an ‘isochoric’ calculation as far as alphaMELTS is concerned because melting is still allowed to cause expansion; this option only controls how much melt must be extracted to *return* to the original volume and, if necessary, also adjusts pressure (for isothermal calculations) or temperature (for isobaric calculations) to maintain equilibrium.

ALPHAMEELTS_FRACTIONATE_SOLIDS

Values: any

Default: unset

To turn on fractional crystallization of all solid phases, set this option to true (does not include water, see below). Do not use this option if you wish to selectively fractionate just a few phases; instead put ‘Fractionate: phase’ lines in your *melts_file* (see the ‘[MELTS file](#)’ section) or adjust individual phase settings with menu option 8.

ALPHAMEELTS_MASSIN

Values: $0 \leq \text{float} < 1$

Default: 0.001 (or 0.0001 for water)

Set to the mass in grams of each solid phase that is retained during fractional crystallization. An increased value may help to stabilize the calculation. A smaller value can also be used, but a minimum of 10^{-6} grams is recommended. Once the phase is no longer in the equilibrium assemblage it will be completely exhausted, regardless of the ALPHAMEELTS_MASSIN value.

ALPHAMEELTS_FRACTIONATE_WATER

Values: any

Default: unset

To remove free water at each calculation stage in an analogous way to how melt is removed during continuous melting, set this variable true. (Note that, in MELTS and pMELTS, water is treated like a ‘solid’, in the sense that it is not melt, so you can achieve the same effect by putting a ‘Fractionate: water’ line in your *melts_file*. However, water is treated differently from the other mineral phases in that it may be extracted during melting or crystallization.)

ALPHAMEELTS_MINW

Values: $0 \leq \text{float} < 1$

Overrides: ALPHAMEELTS_MASSIN

Default: unset

Set to the proportion of retained water, relative to the total system mass. Works in a similar way, for water, as ALPHAMEELTS_MINF does for melt; may be set to exactly zero. If not set then fractionation of water is treated in a similar way to fractionation of a solid phase i.e. a nominal mass of 10^{-4} grams is retained at each stage to stabilize the calculation; the smaller mass of water retained, compared to other solid phases, reflects its significantly lower molecular mass.

ALPHAMEELTS_FRACTIONATE_TARGET

Values: any

Default: unset

During normal forward fractional crystallization a (negative) increment is added to the temperature at each step, ALPHAMEELTS_DELTAT, and the run is terminated when the temperature goes below ALPHAMEELTS_MINT. If this environment variable is set then forward or backward fractionation is performed so that the MgO content (in wt %) or the Mg# (in mol %) hits a particular target; see the next two entries. When option 3 is called alphaMELTS will perform a single calculation for the current *P-T* conditions, in the normal way. When option 4 is used it will first try to find the liquidus instead and use that to decide whether forward or backwards fractionation is required to move the liquid towards the target composition.

For forward fractionation, temperature will be reduced, by an amount equal to the absolute value of ALPHAMEELTS_DELTAT, each time. For backwards fractionation the program will step down (by the same temperature increment) until one or more 'allowed' solid phases join the assemblage. The 'allowed' solid phases are any MgO-bearing phases that have been 'allowed' by setting ALPHAMEELTS_FRACTIONATE_SOLIDS or by having 'Fractionate:' lines in the *melts_file*. The routine then assimilates these phases before searching for the new liquidus. For H₂O-rich compositions, the liquidus temperature will be for a melt composition that is water-saturated, but not oversaturated enough to exsolve vapor (it may also help to buffer *aH₂O*).

Output will be written each time the new liquidus is found. Forward- or backward-fractionated trace element compositions will be calculated if ALPHAMEELTS_DO_TRACE is on. Execution continues until the target MgO or Mg# is reached or just passed; therefore, the smaller ALPHAMEELTS_DELTAT is, the closer the liquid composition will be to the target. In the limit as the step size tends to zero a liquid composition that is on, say, the plagioclase + clinopyroxene cotectic would be expected to stay on the cotectic as back fractionation proceeds (at least until a thermal divide is encountered); in practice one or other solid phase may occasionally be dropped from the assemblage. For complicated peritectic relationships the 'Amoeba' routine (menu option 17) is more useful for constraining a plausible, but still non-unique, parental melt.

ALPHAMEELTS_MGO_TARGET

Values: float

Default: 8.0

Overridden by: ALPHAMEELTS_MGNUMBER_TARGET

Sets the target MgO content of the liquid for forward or backward fractionation to the value in wt %. When using 'Amoeba' (menu option 17) the value of ALPHAMEELTS_MGO_TARGET is used to set the MgO of the *parental* liquid composition and the target MgO content for the *evolved* liquid composition is taken from the *melts_file(s)*. Once 'Amoeba' is finished, reading in

another *melts_file* will revert to using ALPHAMEELTS_MGO_TARGET as the stop point for forward or backward fractional crystallization.

ALPHAMEELTS_MGNUMBER_TARGET

Values: $0 < \text{float} < 100$ Overrides: ALPHAMEELTS_MGO_TARGET

Default: unset

Sets the target Mg# of the liquid for forward or backward fractionation to the value in mol %. When using 'Amoeba' the value of ALPHAMEELTS_MGNUMBER_TARGET is used in a comparable way to ALPHAMEELTS_MGO_TARGET above.

ALPHAMEELTS_ASSIMILATE

Values: any

Default: unset

This environment variable causes a user-defined mass of a second bulk composition to be added after each calculation stage (see '[Bulk composition](#)'). It is intended for calculations at specified *P-T* conditions (e.g. isobaric, isothermal or PTpath modes) or for heat-balanced assimilation in isenthalpic mode. It also works for isentropic or isochoric constraints under certain circumstances but these options are under development and, as yet, untested. Melt may be extracted or solid phases fractionated simultaneously.

On execution, if the mode is isothermal or ALPHAMEELTS_DELTAT is zero *and* the mode is isobaric or ALPHAMEELTS_DELTAP is zero then you will be asked the number of iterations you wish to perform. The program will then request the file type / number of files and the name(s) before the first equilibration⁸. The assimilant bulk composition may be fixed by a single *enrichment_file* or binary *restart_file*, or by providing separate *enrichment_files* for each mineral phase in the assimilant. For separate *enrichment_files*, phase compositions are given in wt% oxides and it is up to the user to ensure the solid compositions are close to stoichiometric in the appropriate pure phase or solid solution model (see the forum for a list of mineral phase end members). Trace element concentrations should be the bulk assimilant values rather than individual mineral ones. A single liquid composition can be input instead of, or in addition to, mineral compositions. Alternatively, multiple *enrichment_files* can be requested so that the assimilant bulk composition can be changed for each iteration. To use multiple *enrichment_files* in this way, see the '[Table file](#)' section for the filename format. The program assumes that the indices are reset each time option 4 is called.

You can enter the mass of assimilant to be added at each subsequent stage or, for *melts_file(s)* only, take the value(s) from the 'Initial Mass:' line(s). If the mass of assimilant is specified for separate mineral *melts_files* then the value entered is the *total* mass and the 'Initial Mass:' lines will be used to determine the proportions of the mineral phases. Major and, if appropriate, trace elements will be mixed (to mix trace elements only use ALPHAMEELTS_FLUX_MELTING). If trace element calculations are switched on, the *enrichment_file(s)* must contain the same trace elements as the *melts_file*. For separate mineral files, the trace elements can be included in each

⁸ To complete one calculation, including any fractionations, before assimilating material choose the separate (liquid and mineral files option the first time. Give a mass (the value does not matter) and then choose 'x' instead of entering any phases. The program will ignore the 'Failure in Remixer' once and ask for the file(s) again at the next iteration.

mineral file, to avoid read errors, but only the values from the first *enrichment_file* will be used. If running in isothermal mode, or the like, then the P , T and fO_2 buffer from the assimilant file will be ignored and set to the current values, regardless of the input file type.

For isenthalpic mode: please set ALPHAMELTS_DELTAP to zero, for reasons given in ‘[Thermodynamic Path](#)’, and ALPHAMELTS_DELTAT to zero so that the number of iterations can be controlled. We suggest you set ALPHAMELTS_SAVE_ALL so that you can gradually build up the number of iterations. In this mode, if you use a single text *enrichment_file* (or files) for the assimilant then the program will try to find a thermodynamically equilibrated state, using a superliquidus start. If you use separate mineral phase *enrichment_files* the enthalpy of each phase is calculated at temperature given in the first *enrichment_file*, without re-equilibration; this is similar to the GUI assimilation routines. If you use a binary assimilant file then it will use the previously calculated starting conditions but the pressure in the file must match the current pressure (or the temperatures must match for isochoric mode). If this is not the case an error message will be printed; either (a) redo the *restart_file* for the correct conditions or (b) use menu option 2 to set the current values to those used to generate the *restart_file* and then call option 4 again.

ALPHAMELTS_FLUX_MELTING

Values: any

Default: unset

This variable causes a user-defined proportion of a second composition, trace elements only, to be mixed in after each calculation stage (see ‘[Bulk composition](#)’). If used in conjunction with ALPHAMELTS_CONTINUOUS_MELTING and ALPHAMELTS_DO_TRACE_H2O it can simulate flux melting by a hydrous fluid (to simulate fluxing by a metasomatic melt use ALPHAMELTS_ASSIMILATE instead). We suggest you set ALPHAMELTS_SAVE_ALL so that you can gradually build up the number of iterations, e.g. until the system achieves steady state, by repeated calling of the ‘execute’ menu option.

Execution is essentially the same as for ALPHAMELTS_ASSIMILATE but for trace elements only. As the mass of the enriching composition is not necessarily defined, the user is asked instead for the mass proportion of the new composition to add (similar to ‘Source Mixer’ in menu option 12). Alternatively, for *melts_files(s)*, the ‘Initial Mass:’ lines may be used; in which case the old and new compositions are mixed in the ratio *old_mass:new_mass*. Note that, as major elements are not mixed, the total system mass (i.e. *old_mass*) will be unaffected.

ALPHAMELTS_DRY_ITER_PATIENCE

Values: $0 \leq \text{integer} \leq 100$

Default: 100

If simulating flux melting or assimilation, this is the maximum number of consecutive iterations that alphaMELTS will run without any melting occurring before it will give up and return to the menu.

pHMELTS and Trace Elements

ALPHAMEELTS_DO_TRACE

Values: any

Default: unset

Implements attached trace element partitioning function for those elements listed in the *melts_file*.

ALPHAMEELTS_DO_TRACE_H2O

Values: any

Default: unset

For the case where water is to be treated as a trace element this option adds an iteration on the H₂O content, as described in *Asimow, et al.* [2004].

ALPHAMEELTS_HK_OL_TRACE_H2O

Values: any

Default: unset

By default the *Mosenfelder, et al.* [2005] model for water solubility in olivine is used. This environment variable uses the *Hirth and Kohlstedt* [1996] model instead, which gives lower solubility and consequently lower partition coefficients.

ALPHAMEELTS_HK_PXGT_TRACE_H2O

Values: 'mineral-melt' or 'mineral-mineral'

Default: mineral-melt

If the *Mosenfelder, et al.* [2005] model for water solubility in olivine is used then by default the mineral-melt partition coefficients for water with orthopyroxene, clinopyroxene and garnet are still those of *Hirth and Kohlstedt* [1996]. Setting this variable to 'mineral-mineral' means the solubility of water in opx, cpx and garnet are linked to the newer water solubility in olivine model, which is equivalent to preserving the mineral-mineral water partition coefficients of *Hirth and Kohlstedt* [1996]. Setting it to 'mineral-melt' retains the default behavior.

ALPHAMEELTS_2X_OPX_TRACE_H2O

Values: any

Default: unset

By default the solubility of water in clinopyroxene is twice that in orthopyroxene, based on the results of *Hirth and Kohlstedt* [1996]. If this variable is set then the solubility of water in opx is scaled up by to be equal to the cpx value, consistent with the observations of *Hauri et al.* [2006]. This option can be used regardless of whether the *Mosenfelder, et al.* [2005] or *Hirth and Kohlstedt* [1996] models are being used for olivine or opx, cpx and garnet. For example, if either ALPHAMEELTS_HK_OL_TRACE_H2O or ALPHAMEELTS_HK_PXGT_TRACE_H2O is set then the mineral-mineral partition coefficients will be those from Table 1 of *Hirth and Kohlstedt* [1996], except that olivine-opx value of 0.2 will be replaced by 0.1.

ALPHAMELTS_TRACE_DEFAULT_DPTX

Values: any

Default: unset

By default all partition coefficients used in trace element calculations are constant. If ALPHAMELTS_TRACE_VARIABLE_D is set then $D = D(P,T,X)$ are calculated for elements and phases in Table 1. Constant partition coefficients will be used for all other elements / phases. This list may be modified in the *trace_data_file*, as previously explained.

ALPHAMELTS_TRACE_NORMALIZATION

Values: $1 \leq integer \leq 4$

Default: unset

If set, this chooses one of four compositions to normalize trace elements to (if any):

if integer is 1, Primitive Mantle of *Sun and McDonough* [1989];

if integer is 2, DMM of *Workman and Hart* [2005];

if integer is 3, Primitive mantle of *McKenzie and O'Nions* [1991; 1995];

if integer is 4, Depleted mantle of *McKenzie and O'Nions* [1991; 1995].

Sample input files showing concentrations for each of the idealized source compositions above are provided as illustrations; note that some elements (i.e. Ni, Cr, and Mn) appear as major and trace elements because their inclusion in the liquid calibration differs between MELTS and pMELTS. Isotopes are normalized to the 'non-isotope' abundance (see the '[MELTS file](#)' section). This option is useful if the source composition given in the *melts_file* is different from the four options above. If you wish to normalize to the source in the *melts_file* the simplest thing is to provide the list of elements with '1.0' for each of the abundances (except isotopes).

ALPHAMELTS_TRACE_INPUT_FILE

Values: filename

Default: unset

Gives the name of the *trace_data_file*, which may be used to change partition coefficients, determine whether variable partition coefficients are calculated and with which parameters, as described above.

ALPHAMELTS_TRACE_USELIQFEMG

Values: any

Default: unset

By default the Mg# of the melt, needed to estimate $D(P,T,X)$ for clinopyroxene, is estimated from the clinopyroxene composition using Equation 35 from *Wood and Blundy* [1997]). If this environment variable is set then the Mg# of the melt is taken directly from the (pH)MELTS calculated liquid composition. If no liquid is present then the program will revert to the default behavior and use the clinopyroxene composition.

Input options

ALPHAMEELTS_ADIABAT_BIN_FILE

Values: any

Default: unset

Unavoidable changes within the programs mean that binary *restart_files* created with version 2.0+ of Adiatat_1ph will not read into alphaMELTS. Once you have read in the file, if you immediately save it again it will be updated to Adiatat_1ph 3.0 format. Adiatat_1ph 3.0 format *is* compatible with alphaMELTS 1.0+ so unset this environment variable for subsequent runs. For more details see ADIABAT_V20_BIN_FILE in the Adiatat_1ph 3 documentation.

Output options

ALPHAMEELTS_CELSIUS_OUTPUT

Values: any

Default: unset

By default, temperature input to alphaMELTS is in °C, whereas temperature output is in Kelvin. When this environment variable is set, text file and screen output is also in °C.

ALPHAMEELTS_SAVE_ALL

Values: any

Default: unset

By default, the main output file is only written for calculations made in the most recent call to menu option 4. If this variable is set then results from all calculations are saved and output. This provides a simple way to record single calculations (menu option 3) or to build up results from multiple iterations or multiple *melts_files*. Note that even if the appropriate environment variables are set, no solid or liquid fractionation will occur until menu option 4 is run. If using ‘Amoeba’ (menu option 17) the ALPHAMEELTS_SAVE_ALL function will be temporarily switched off (so that iterations of the search algorithm are not saved); when it is finished a single calculation (menu option 3) may be used to write output for the best-fit *parental* melt.

ALPHAMEELTS_SKIP_FAILURE

Values: any

Default: unset

Normally failure of the minimization routines means that the `alphamelts` executable must be restarted to clear the memory. If this environment variable is set then a copy of the thermodynamic state is made each time a successful calculation is made. If the next calculation fails, this last good state (or the bulk composition and starting conditions from the *melts_file*, if it is the first calculation) is used for subsequent attempts. This may be useful if you are very close to a reaction when the algorithms may have trouble deciding whether to add a phase to the assemblage and need to overstep just slightly. It is also helpful when trying out starting solutions using options 2 and 3. If ALPHAMEELTS_SAVE_ALL is also set, the last good state will also be

written to the output files as a placeholder; it should be obvious that this represents a skipped failure as all values, including pressure and temperature, will be identical to the previous output.

ALPHAMEELTS_FAILED_ITER_PATIENCE

Values: $0 \leq \text{integer} \leq 10$

Default: 10

If the minimization routines do not recover in the next few iterations then one drawback of ALPHAMEELTS_SKIP_FAILURE is that it can lead to infinite loops of failure or, if triggered repeatedly, eventually to a segmentation fault. If ALPHAMEELTS_SKIP_FAILURE is set, this variable is the maximum number of consecutive failed iterations that alphaMELTS will run before a clean return to the menu where parameters, such as the choice of fO_2 buffer, can be adjusted or alphaMELTS completely restarted. If running in PTgrid or TPgrid modes, it will be the maximum number of failed iterations before alphaMELTS moves to the next internal loop (T -loop or P -loop respectively).

ALPHAMEELTS_INTEGRATE_FILE

Values: filename

Default: unset

Normally the *integrated_output_file* can be written immediately after execution finishes, but if `alphameelts` tends to crash before the menu option 16 can be called then memory contents will be lost. As a precaution, ALPHAMEELTS_INTEGRATE_FILE should be set to record details of the packets of melt extracted during melting to a file. If `run_alphameelts.command` is restarted with the same settings and menu option 16 is called before execution then the resulting file will be used as input for the melt integration. Note that ALPHAMEELTS_INTEGRATE_FILE is not the *integrated_output_file* itself but an intermediate file that may be used in its generation.

ALPHAMEELTS_LATENT_HEAT

Values: any

Default: unset

Normally text file output is done just after equilibration and before any fractionation. If ALPHAMEELTS_LATENT_HEAT to set to 'true' `alphameelts` will also output the state just after fractionation and just before the next equilibration (when the temperature has been lowered and the system cooled but not allowed to crystallise any more yet). This gives a simple way to calculate the latent heat of crystallization, without having to do the cooling adjustment described in *Ghiorso* [1997]. Note that, for the same MORB composition, the results will differ from Figure 5 of *Ghiorso* [1997]. This apparent discrepancy is due to changes in the MELTS software since it was first released and not to differences in how the variation in latent heat is accessed.

ALPHAMEELTS_QUICK_OUTPUT

Values: any

Default: unset

If this environment variable is set then equilibrated thermodynamic states from previous calculations are not saved, except as required for ALPHAMEELTS_SKIP_FAILURE etc. For

calculations comprising a large number of iterations, such as those on a closely spaced P - T grid, setting this variable can significantly reduce memory usage and therefore increase computational speed as the calculation proceeds. Reporting of fO_2 will be affected slightly (see ‘[Main output files](#)’). The `Phase_mass_tbl.txt` and `Phase_vol_tbl.txt` files will not be written, which will cause `run_alphamelts.command` to issue a warning about not being able to find these files; the warning can be safely ignored.

Others

ALPHAMELTS_MULTIPLE_LIQUIDS

Values: any

Default: unset

This turns on exsolution of immiscible liquids. The solvi are not very well determined, and we do not recommend serious use of this feature, but in some cases operation inside an unrecognized two-liquid field can lead to path-dependent non-unique equilibria. This option should not be used if trace element calculations are enabled.

ALPHAMELTS_FRACTIONATE_SECOND_LIQUID

Values: any

Default: unset

When running with `ALPHAMELTS_MULTIPLE_LIQUIDS` set, treat all liquids except the first as fractionating phases and remove them from the system after each equilibration.

ALPHAMELTS_FOCUS

Values: any

Default: unset

Option to do the focusing calculation described in *Asimow and Stolper* [1999]. It works by multiplying the mass of liquid in system by a fixed factor after each equilibration.

ALPHAMELTS_FOCUS_FACTOR

Values: integer or float

Default: unset

When `ALPHAMELTS_FOCUS` is set, this determines the multiplication factor for the mass of liquid. Usually it will be a number slightly greater than unity, like the 100th root of 2. If `ALPHAMELTS_FOCUS` is set without a value in `ALPHAMELTS_FOCUS_FACTOR` the focusing calculation loop will be ignored.

ALPHAMELTS_INTEGRATE_PHI

Values: any

Default: unset

Normally, packets of extracted melt are saved in memory (and, optionally, written out to the `ALPHAMELTS_INTEGRATE_FILE` ‘crash’ file, described above) by mass i.e. in terms of dF. If

this variable is set then the packets of extracted melt will be saved by volume i.e. in terms of dPhi. As the full pressure integral (calculation of crustal thickness etc.) is not strictly correct for increments in Phi, that option will not be offered. However, the opportunity to interpolate for equal increments of melt fraction by volume will be available.

The `column_pick.command` script

Synopsis

```
column_pick.command column_list_file [> [table_file]]
```

The *column_list_file* (see below) can actually be replaced by a list of such files, separated by spaces, in which case the script will act as if all the *column_list_files* had been concatenated into one. If you are using the double-click method then, when prompted, you can type the full '*column_list_file(s) > table_file*' part or you can enter one or more *column_list_files*, separated by spaces, in which case the script will ask for the name of the *table_file* later. You need to give the full path to *table_file*, or the output file will end up in the *links_folder* folder, so you may want to make an empty file with the same name that you can drag-and-drop.

Omitting '*> table_file*' causes all output to be printed to the screen and will only be useful to diagnose an error when calling the script from Matlab. Likewise, using '>' with no *table_file* will cause an error, except in Matlab. Even then there still needs to be a space after the '>', e.g.:

```
meltsdata = perl('column_pick.command', 'columns.txt', '> ')
```

Including this last argument means that warnings (e.g. 'Wrote output') are not printed so that the returned block of data may be passed to Matlab's 'textscan' function. The 'column_pick.m' example shows one way to automate calling `column_pick.command` and processing the output; it can easily be tailored to the user's specific needs. Type 'help column_pick' for more information.

Column list file

The instructions that `column_pick.command` reads from the *column_list_file* are mostly sets of File-Table-Variable or File-Table-Columns triplets of lines (must be in that order) repeated as many times as necessary. 'File' may be any of the files written by alphaMELTS⁹. Other valid lines are shown in the example below.

If you are using `column_pick.command` in double-click mode then generally the script will search the *links_folder* for files in the 'File: *filename*' lines. Alternatively, use the 'Name: *filename*' line beforehand, where *filename* is the name of *column_list_file* (not including its path). In that case the script will look for the named files in the same folder as the *column_list_file* or relative to it. If you are using the '-c *column_file*' switch of `run_alphamelts.command` you should not need the 'Name: *filename*' line as `column_pick.command` is called before the output files are moved the folder defined by '-p *path*', if any. Within the *column_list_file* any filenames must be in plain text, i.e. without special characters being escaped, and names containing spaces must be put in quotes.

⁹ This includes thermodynamic output from option 15, though 'grep' or the Windows equivalent 'findstr' may be more useful tools for that file as it more row-oriented.

For the 'Delimiter:' line, put one of the words 'space', 'comma' or 'tab' or the appropriate character ('\t' for tab), with or without single or double quotes. The default, if there is no 'Delimiter:' line, is to use 'space'. If you are picking compositional information (major or trace elements) for a solid phase that is not present at all P , T conditions in the run then lines where it is absent will be padded with whichever delimiter was chosen.

Two different formats can be chosen for the header line(s) in the output. If 'Header: Excel' used, which is the default, the full table names (e.g. "Phase Masses") will appear in the first row padded in such a way to align correctly in Excel. Column names will be in the second row and values in the rest of the file. In Excel, select the correct delimiter and make sure 'Treat consecutive delimiters as one' is not selected. If 'Header: Matlab' is chosen then there will be a single header line of '*columnname_tablename*', where '*tablename*' is the short form of the title (e.g. 'phase').

```
! An exclamation mark can be used to comment out all or part of a line
Name: columns.txt      ! used to deduce the input path, e.g. in double-click mode
!Delimiter: \t        whole line is commented out so will default to 'space'
Header: Matlab         ! case shouldn't matter so 'Matlab' or 'Matlab' is fine

File: System_main_tbl.txt      ! first file must have an entry for each recorded step
Table: System
Columns: Pressure Temperature F
File: Phase_mass_tbl.txt      ! the variables are currently unset ...
Table: "Phase Masses"
Columns: liquid_0 garnet_0 clinopyroxene clinopyroxene_1
File: Liq_comp_tbl.txt
Table: Liquid
Columns: SiO2 MgO
File: Phase_mass_tbl.txt
Table: Phase
Variables: Pressure Temperature garnet_0
... continued from previous page ...
File: Phase_main_tbl.txt
Table: garnet                ! this is the same as putting 'garnet_0'
Columns: mass CaO
File: Trace_main_tbl.txt
Table: garnet_0             ! the variables are still set to 'Temperature Pressure garnet_0'
Columns: mass La
Variables:                   ! this unsets the variables, we don't need them for the next file
File: System_main_tbl.txt
Table: System
Columns: phi
... repeat groups of three lines as many times as needed for all tables ...
```

The script makes very few assumptions about the format of the files it processes, except that column names must not contain spaces and it is unable to distinguish between two columns of the same

name. Where appropriate, `run_alphamelts.command` replaces spaces with ‘_’ in the raw `alphamelts` output and all solution phases are output as ‘*phase_number*’, so even if phase separation occurs there should be no need to rename the columns. Omitting the trailing underscore and number on ‘*phase*’ is synonymous with ‘*phase_0*’ but in the output it will appear without the ‘_0’. There is also no need to rename any ‘fO2(*buffer*)’ columns in ‘System_main_tbl.txt’ as identical titles on the columns imply that the corresponding entries are also identical. The ‘Table:’ line is followed by the first word(s) of the title of the table you wish to pick out. The first table processed should have a row for every recorded equilibrium state; else the script’s behavior will be unpredictable. Use single or double quotes to contain table names that contain spaces. Case should not matter but the column names must match exactly in all other respects (e.g. ‘pressure’ not ‘P’). There is no requirement that columns be listed in the same order as in the original file. If a column is not found the script continues with a warning and the space padded with the appropriate delimiter; this makes it easy to concatenate results from different runs that do not necessarily include a particular phase.

The ‘Variables:’ line is used to read in between 0 and 3 columns of a table, usually one with a row for each recorded equilibrium state. These columns of values will be used to check and align the chosen columns from subsequent tables, either by matching the variable and column names exactly, or by matching the variable name and the table name in the sense described above (e.g. ‘clinopyroxene_0’ column in ‘Phase_mass_tbl.txt’ matches ‘clinopyroxene’ table in ‘Phase_main_tbl.txt’) and assuming the target column for comparison is the ‘mass’ one. Where a solid or liquid phase is missing, the row will be padded accordingly. Simple calculations, e.g. isentropic or isobaric, will require ‘Pressure’ or ‘Temperature’ as a variable, whereas runs in ‘PTpath’ mode will often require ‘Pressure’ and ‘Temperature’. If the calculation tends to go over the same P - T conditions repeatedly e.g. an isobaric-isothermal flux melting calculation then include the ‘mass’ variable as well. ‘Phase_mass_tbl.txt’ will be the most convenient file to set variables from in practice.

The variable set, or lack thereof, will persist until the next ‘Variables:’ line. You can ‘unset’ all variables by leaving the rest of the line blank; in this case the ‘Variables:’ line can appear on its own instead of in a triplet. Setting the variables is optional for tables like the one in ‘Liquid_comp_tbl.txt’ that do not have any missing rows but it will not necessarily cause an error (as ‘liquid_0’ will match ‘Liquid’). If all the files used have the full compliment of rows then the resulting output file can actually be used as one of the input files for another run of `column_pick.command`; this may require minor edits to ensure the table has a title and / or all column names are distinct. Output from `column_pick.command` can also be used as input to the `table_file` function of `run_alphamelts.command` so that quite complex scenarios can be modeled.

What to do when (pH)MELTS misbehaves

Although the algorithms used to calculate equilibria are powerful, there are numerous situations where they will fail to converge to a solution and a few situations where they will apparently converge to an incorrect solution. We advise persistence; strategies are available for dealing with these situations, though they may require some patience.

- Sometimes (pH)MELTS does not converge. Try varying parameters or approaching the desired condition by smaller steps in the intensive variables; you may need to restart `alphamelts` to give the algorithm a suitable starting solution again. For example, when doing isentropic melting calculations, it may help to reduce the size of the pressure decrement. If you have set `ALPHAMELTS_CONTINUOUS_MELTING`, then try increasing

ALPHAMELTS_MINF slightly (in particular, having it set to 0 can make convergence tricky). Similarly, adjusting ALPHAMELTS_MINW or ALPHAMELTS_MASSIN may be beneficial if you are fractionating water or solid phases.

- Often, the first calculation in a sequence is the hardest because the initial guess is so far from the true equilibrium. Try creeping up on the desired condition in smaller steps. Often it is best to start at or above the liquidus and lower the temperature to the desired point. For example, if you require a cold subsolidus start and the composition is neither peridotite nor basaltic then the norm routine may choke if you try to reach those conditions in one step.

In the particular case that (pH)MELTS is not converging on the initial calculation you could also try the following procedure: (1) Find some other set of P , T conditions for which it will converge; (2) Work towards the desired starting conditions using a combination of menu options 2, 3 or 4; (3) Optionally, save a *restart_file* (or files), using option 13, so that future runs can be started at or near the desired conditions. See menu option 2 for more details.

- Another source of difficulty is fO_2 -buffered calculations, especially below the solidus. For a sub-solidus starting solution: try turning the oxygen buffer off, obtaining an equilibrium solution, and then turning the buffer back on (use menu option 5). Again, you may like to use option 13 to save a *restart_file* (or files), which can be read in later with option 11.
- It is possible that the minimization algorithm in `alphamelts` may become trapped in a local minimum and will converge to an incorrect solution. Sometimes such false equilibria can be averted by suppressing a competing phase (see menu option 8). More often, problems occur because the relative stability of the pyroxenes is determined by quite small energy differences and the wrong pyroxene can appear and trap the calculation in a metastable state. Version 1.8 of `Adiabat_1ph` introduced a workaround that should help alleviate this problem to some extent. However, if you suspect this has arisen, try manually increasing the temperature until the ‘wrong’ pyroxene disappears and then approaching the desired P , T condition from a different direction to the first attempt. If all else fails, try the *restart_file* workaround described in ‘[Restart file use: phase metastability](#)’.
- Another source of non-uniqueness and path dependence in calculations can be operating inside a two-liquid miscibility gap. The ALPHAMELTS_MULTIPLE_LIQUIDS option is available to deal with this situation, although the liquid solvi are not well calibrated and handling multiple liquids in a variety of situations such as fractional melting, trace element calculations, etc., can be complicated. We recommend the multiple liquids option only as a last resort when odd behavior is caused by not recognizing the metastability of the liquid.

Bugs and fixes

- Note that `run_alphamelts.command` only passes your chosen settings to `alphamelts` indirectly – either by setting environment variables or by making entries in text files. If you are in batch or automatic mode it will redirect the `alphamelts` standard input by executing ‘`alphamelts < batch_file`’, which passes the *melts_file* name. Otherwise, the user must still prompt `alphamelts` to read in the relevant text or binary files (this is true even when the ‘-m’ switch is used); a warning will be given if no file has been input.
- The error message ‘No calculations performed’ usually means that calculations are being attempted outside the P or T limits set by the environment variables. Note that the default maximum pressure is 4 GPa for pMELTS and 3 GPa for MELTS calculations.

- The incorporation of Cr₂O₃ in solid phases in MELTS is oversimplified. If this component is included in the bulk composition then the range of conditions at which spinel may crystallize will be vastly overestimated to the point of its being almost ubiquitous (see *Asimow et al.* [1995] for more details). If Cr₂O₃ is omitted spinel stability will be underestimated. Other limitations in the MELTS, pMELTS and pHMELTS algorithms have been acknowledged and discussed extensively elsewhere (e.g. *Hirschmann, et al.* [1998]).
- Security features in later versions of Windows have also been known to disable Perl's ability to read the command line input, such as switches or file names. The `install.command` script should take care of this issue during set up so if the '.command' scripts stop working correctly (e.g. after a Service Pack is installed) try rerunning `install.command`.
- On Mac OS X, if 'bash' is not the default shell or the path is not properly set the user may find that double-click mode for `run_alphamelts.command` does not work. One remedy is to make a copy of the `alphamelts_macosx32` (or 64) executable in the `installation_folder` and rename it simply 'alphamelts'.
- If you get a message like '`file_name`: Permission denied' in Linux or Mac OS X then make sure that `alphamelts` and / or `run_alphamelts.command` are executable. Try typing '`chmod 755 file_name`' at the terminal command line to make sure.

Also, if Perl is installed somewhere other than `/usr/bin` then the scripts will not work but you can edit the first line of each Perl file and replace '`/usr/bin`' with the correct location. Try typing '`which perl`' at the command line to find the installation path.

- Occasionally `run_alphamelts.command` produces warning messages like 'Uninitialized variable in string, line...'. Such messages are usually harmless but may be symptomatic of a more serious problem (e.g. `alphamelts` did not behave in the expected fashion).
- If the terminal setup is incorrect you may find that the 'backspace' key does not work. This is most likely to occur if you are running alphaMELTS remotely, via `ssh`. `Ctrl-H` may work instead. Otherwise it may be more convenient to use a `batch_file` for input.
- When you start the `alphamelts` program then (if it has been more than 6 hours) it will check the version against the <https://magmasource.caltech.edu/alphamelts/> server and tell you if you need to update to a newer one. If a connection cannot be established for any reason you may notice a pause, up to ~3 seconds, before the program messages and menu are displayed.
- Bugs that have been fixed and features that have been added to the code are listed briefly in the online file 'README.TXT' and the documentation has been updated to include these and to clarify parts, based on users' experiences. If you find a problem that is not listed in this manual, please check that you have the latest version of the software first and then contact us.

Disclaimer

The pHMELTS algorithm and alphaMELTS software package are the work of Paul Asimow (asimow@gps.caltech.edu) and Paula Antoshechkina (psmith@gps.caltech.edu). All questions, complaints and suggestions for improvement in the software or the documentation are welcome, and may be posted at <https://magmasource.caltech.edu/forum/> or sent to Antoshechkina (particularly Perl, output, and front-end issues) or Asimow (pHMELTS model or algorithm questions). These programs come with no warranty or explicit promise of support, but the authors will gladly answer questions

and provide reasonable support when possible. Please note that our ability to help will be greatly increased if you describe the calculation being attempted in some detail. Useful information includes, but is not limited to: what you typed at the terminal command line and `alphamelts` menu prompt; the `melts_file` and `settings_file` you are using; what messages or output you got, if any (for example, try copying the screen output to a text file: on Linux, use the third mouse button; on Mac OS X, command-C and command-V work for copy and paste; on Windows, make sure 'Quick Edit Mode' is on in `cmd.exe` and then use the right mouse button to copy and paste). As with any numerical model, caveat calculator: do not believe results beyond the calibrated range and sanity check all results when possible. If you must base expensive or dangerous experiments or industrial operations on this model, we will not be responsible for the consequences.

Bibliography

- Antoshechkina, P. M., P. D. Asimow, E. H. Hauri, and P. I. Luffi (2010), Effect of water on mantle melting and magma differentiation, as modeled using `Adiabat_1ph 3.0`, Abstract V53C-2264 presented at 2010 Fall Meeting, AGU, San Francisco, Calif., 13-17 Dec.
- Asimow, P. D., J. E. Dixon, and C. H. Langmuir (2004), A hydrous melting and fractionation model for mid-ocean ridge basalts: Application to the Mid-Atlantic Ridge near the Azores, *Geochim. Geophys. Geosyst.*, **5**, Q01E16, doi:10.1029/2003GC000568.
- Asimow, P. D. and J. Longhi (2004), The significance of multiple saturation points in the context of polybaric near-fractional melting, *J. Petrol.*, **45**, 2349-2367.
- Asimow, P. D., and M. S. Ghiorso (1998), Algorithmic modifications extending MELTS to calculate subsolidus phase relations, *Am. Mineral.*, **83**, 1127-1132.
- Asimow, P. D., M. M. Hirschmann, M. S. Ghiorso, M. J. Ohara, and E. M. Stolper (1995), The Effect of Pressure-Induced Solid-Solid Phase-Transitions on Decompression Melting of the Mantle, *Geochim. Cosmochim. Ac.*, **59**, 4489-4506.
- Asimow, P. D., M. M. Hirschmann, and E. M. Stolper (2001), Calculation of peridotite partial melting from thermodynamic models of minerals and melts, IV. Adiabatic decompression and the composition and mean properties of mid-ocean ridge basalts, *J. Petrol.*, **42**, 963-998.
- Asimow, P. D., and E. M. Stolper (1999), Steady-state mantle-melt interactions in one dimension: I. Equilibrium transport and melt focusing, *J. Petrol.*, **40**, 475-494.
- Berman, R. G. (1988), Internally-consistent thermodynamic data for minerals in the system Na₂O-K₂O-CaO-MgO-FeO-Fe₂O₃-Al₂O₃-SiO₂-TiO₂-H₂O-CO₂. *J. Petrol.*, **29**, 445-522.
- Berman, R. G. (1990), Mixing Properties of Ca-Mg-Fe-Mn Garnets, *Am. Mineral.*, **75**, 328-344.
- Berman, R. G., and A. M. Koziol (1991), Ternary Excess Properties of Grossular-Pyrope-Almandine Garnet and Their Influence in Geothermobarometry, *Am. Mineral.*, **76**, 1223-1231.
- Blundy, J., and B. Wood (1994), Prediction of Crystal-Melt Partition-Coefficients from Elastic-Moduli, *Nature*, **372**, 452-454.
- Blundy, J. D., T. J. Falloon, B. J. Wood, and J. A. Dalton (1995), Sodium Partitioning between Clinopyroxene and Silicate Melts, *J. Geophys. Res.*, **100**, 15501-15515.
- Cooper, K. M., J. M. Eiler, P. D. Asimow, and C. H. Langmuir (2004), Oxygen isotope evidence for the origin of enriched mantle beneath the mid-Atlantic ridge, *Earth. Planet. Sc. Lett.*, **220**, 297-316.
- Ghiorso, M. S. (1994), Algorithms for the Estimation of Phase-Stability in Heterogeneous Thermodynamic Systems, *Geochim. Cosmochim. Ac.*, **58**, 5489-5501.
- Ghiorso, M. S. (1997), Thermodynamic Models of Igneous Processes, *Annu. Rev. Earth Planet. Sci.*, **25**, 221-241.

- Ghiorso, M. S., and G. A. Gualda (2015), An H₂O–CO₂ mixed fluid saturation model compatible with rhyolite-MELTS, *Contrib. Mineral. Petr.*, **169**, 1-30.
- Ghiorso, M. S., M. M. Hirschmann, P. W. Reiners, and V. C. Kress (2002), The pMELTS: A revision of MELTS for improved calculation of phase relations and major element partitioning related to partial melting of the mantle to 3 GPa, *Geochem. Geophys. Geosyst.*, **3**, 1030, doi:10.1029/2001GC000217.
- Ghiorso, M. S., and R. O. Sack (1991), Fe-Ti oxide geothermometry: Thermodynamic formulation and the estimation of intensive variables in silicic magmas, *Contrib. Mineral. Petr.*, **108**, 485-510.
- Ghiorso, M. S., and R. O. Sack (1995), Chemical Mass-Transfer in Magmatic Processes .4. A Revised and Internally Consistent Thermodynamic Model for the Interpolation and Extrapolation of Liquid-Solid Equilibria in Magmatic Systems at Elevated-Temperatures and Pressures, *Contrib. Mineral. Petr.*, **119**, 197-212.
- Gualda, G. A., M. S. Ghiorso, R. V. Lemons, and T. L. Carley (2012), Rhyolite-MELTS: a modified calibration of MELTS optimized for silica-rich, fluid-bearing magmatic systems, *J. Petrol.*, **53**, 875-890.
- Hauri, E. H., G. A. Gaetani and T. H. Green (2006), Partitioning of water during melting of the Earth's upper mantle at H₂O-undersaturated conditions, *Earth. Planet. Sc. Lett.*, **248**, 715-734.
- Hamecher, E. A., P. M. Antoshechkina, M. S. Ghiorso, and P. D. Asimow (2013), The molar volume of FeO-MgO-Fe₂O₃-Cr₂O₃-Al₂O₃-TiO₂ spinels, *Contrib. Mineral. Petr.*, **165**, 25-43.
- Hebert, L. B., P. Antoshechkina, P. Asimow, and M. Gurnis (2009), Emergence of a low-viscosity channel in subduction zones through the coupling of mantle flow and thermodynamics, *Earth. Planet. Sc. Lett.*, **278**, 243-256.
- Hirschmann, M. M., M. S. Ghiorso, L. E. Wasylenki, P. D. Asimow, and E. M. Stolper (1998), Calculation of peridotite partial melting from thermodynamic models of minerals and melts. I. Review of methods and comparison with experiments, *J. Petrol.*, **39**, 1091-1115.
- Hirth, G., and D. L. Kohlstedt (1996), Water in the oceanic upper mantle: Implications for rheology, melt extraction and the evolution of the lithosphere, *Earth. Planet. Sc. Lett.*, **144**, 93-108.
- Hunziker, P. (2003), The stability of tri-octahedral Fe²⁺-Mg-Al chlorite. A combined experimental and theoretical study, Ph.D. Thesis, University of Basel.
- Kessel, R., J. R. Beckett, and E. M. Stolper (2001), Thermodynamic Properties of the Pt-Fe system, *Am. Mineral.*, **86**, 1003-1014.
- Kress, V. C., and I. S. E. Carmichael (1991), The Compressibility of Silicate Liquids Containing Fe₂O₃ and the Effect of Composition, Temperature, Oxygen Fugacity and Pressure on Their Redox States, *Contrib. Mineral. Petr.*, **108**, 82-92.
- McKenzie, D., and R. K. O'Nions (1991), Partial Melt Distributions from Inversion of Rare-Earth Element Concentrations, *J. Petrol.*, **32**, 1021-1091.
- McKenzie, D., and R. K. O'Nions (1995), The Source Regions of Ocean Island Basalts, *J. Petrol.*, **36**, 133-159.
- McMullin, D. W. A., R. G. Berman, and H. J. Greenwood (1991), Calibration of the SGAM Thermobarometer for Pelitic Rocks Using Data from Phase-Equilibrium Experiments and Natural Assemblages, *Can. Mineral.*, **29**, 889-908.
- Mosenfelder, J. L., N. I. Deligne, P. D. Asimow, and G. R. Rossman (2005), Hydrogen incorporation in olivine from 2-12 GPa, *Am. Mineral.*, **91**, 285-294.
- Myers, J., and Eugster, H. P. (1983), The system Fe-Si-O: Oxygen buffer calibrations to 1500 K, *Contrib. Mineral. Petr.*, **82**, 75-90.
- Plank, T., M. Spiegelman, C. H. Langmuir, and D. W. Forsyth (1995), The Meaning of Mean-F – Clarifying the Mean Extent of Melting at Ocean Ridges, *J. Geophys. Res.*, **100**, 15045-15052.

- Reiners, P. W., B. K. Nelson, and M. S. Ghiorso (1995), Assimilation of Felsic Crust by Basaltic Magma – Thermal Limits and Extents of Crustal Contamination of Mantle-Derived Magmas, *Geology*, **23**, 563-566.
- Sack, R. O., and M. S. Ghiorso (1989), Importance of Considerations of Mixing Properties in Establishing an Internally Consistent Thermodynamic Database – Thermochemistry of Minerals in the System Mg_2SiO_4 - Fe_2SiO_4 - SiO_2 , *Contrib. Mineral. Petr.*, **102**, 41-68.
- Sack, RO, Ghiorso, MS (1991) Chromian spinels as petrogenetic indicators: thermodynamics and petrological applications. *Am. Mineral.*, 76, 827-847.
- Shannon, B. (1976), Revised effective ionic radii and systematic studies of interatomic distances in halides and chalcogenides, *Acta. Crystallogr.*, **32**, 751-767.
- Smith, P. M., and P. D. Asimow (2005), Adiatat_1ph: A new public front-end to the MELTS, pMELTS, and pHMELTS models, *Geochem. Geophys. Geosyst.*, **6**, Q02004, doi:10.1029/2004GC000816.
- Smith, P. M., Baker, L. J., Asimow, P. D., and Gurnis, M. C. (2007), Coupled petrological and geodynamic models of mantle flow in subduction zones; the importance of chlorite in the emergence of a low-viscosity channel, Abstract V43D-1637 presented at 2007 Fall Meeting, AGU, San Francisco, Calif., 10-14 Dec.
- Sun, S. S., and W. F. McDonough (1989), Chemical and isotopic systematics of oceanic basalts; implications for mantle composition and processes, *Geol. Soc. Spec. Publ.*, **42**, 313-345.
- Thompson, R. N., A. J. V. Riches, P. M. Antoshechkina, D. G. Pearson, G. M. Nowell, C. J. Ottley, A. P. Dickin, V. L. Hards, A.-K. Nguno and V. Niku-Paavola (2007), Origin of CFB magmatism: Multi-tiered intracrustal picrite-rhyolite magmatic plumbing at Spitzkoppe, western Namibia, during early-Cretaceous Etendeka magmatism, *J. Petrol.*, **48**, 1119-1154.
- van Westrenen, W., J. Blundy, and B. Wood (1999), Crystal-chemical controls on trace element partitioning between garnet and anhydrous silicate melt, *Am. Mineral.*, **84**, 838-847.
- Wood, B. J., and J. D. Blundy (1997), A predictive model for rare earth element partitioning between clinopyroxene and anhydrous silicate melt, *Contrib. Mineral. Petr.*, **129**, 166-181.
- Wood, B. J., J. D. Blundy, and J. A. C. Robinson (1999), The role of clinopyroxene in generating U-series disequilibrium during mantle melting, *Geochim. Cosmochim. Ac.*, **63**, 1613-1620.
- Workman, R. K., and S. R. Hart (2005), Major and trace element composition of the depleted MORB mantle (DMM), *Earth. Planet. Sc. Lett.*, **231**, 53-72.