INSTRUCTIONS FOR ADIABAT_1PH PROGRAM &
RUN_ADIABAT.PL SCRIPT; TEXT DRIVEN INTERFACES FOR
MELTS, pMELTS & pHMELTS.

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

March 27, 2008 (Version 2.0, revision 1).

Introduction

The program adiabat_lph is a simple text-menu driver for subroutine versions of the algorithms MELTS, pMELTS and pHMELTS [Asimow, et al., 2004; Ghiors, et al., 2002; Ghiors 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 $aH_2O$. 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 described in general in the following section before a detailed list of the input and output files, options and switches and how to use them. For background on the MELTS and pMELTS software (GUI versions), please visit the http://melts.ofm-research.org/ site.

The software is available at http://www.gps.caltech.edu/~asimow/adiabat and is suitable for Linux (32 bit and 64 bit processors), MacOS X (PowerPC or Intel based) and Windows. Many aspects of program execution are controlled by environment variables. The Perl file run_adiabat.pl may be used to invoke adiabat_lph 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.

Description

Thermodynamic model

The underlying thermodynamic model behind the adiabat_lph front end is either MELTS [Ghiors and Sack, 1995] or pMELTS [Ghiors, et al., 2002]. The former is calibrated on a wide variety of bulk compositions in the system SiO$_2$-TiO$_2$-Al$_2$O$_3$-Fe$_2$O$_3$-Cr$_2$O$_3$-FeO-MnO-MgO-NiO-CoO-CaO-Na$_2$O-K$_2$O-P$_2$O$_5$-H$_2$O and is suitable at pressures not greater than 3 GPa. The latter is intended for peridotite compositions only, in the system SiO$_2$-TiO$_2$-Al$_2$O$_3$-Fe$_2$O$_3$-Cr$_2$O$_3$-FeO-MgO-CaO-Na$_2$O-K$_2$O-P$_2$O$_5$-H$_2$O, 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.

In MELTS or pMELTS, H$_2$O is treated as an oxide component, and can partition into melt, pure vapor, or hydrous minerals (including simple amphiboles and micas). In pHMELTS, the effects of storage in water in nominally anhydrous minerals is approximately modeled [Asimow, et al., 2004]. The pHMELTS calculation uses trace element partitioning to distribute H$_2$O between the system and a “hidden” reservoir representing H$_2$O dissolved in nominally anhydrous minerals. The
adiabat_1ph program can run MELTS or pMELTS in their conventional form 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 isochoic but this has not been tested. In this document, MELTS, pMELTS and/or pHMELTS are sometimes collectively called (pH)MELTS.

**Alternative garnet and biotite models**

The Ca-Mg-Fe garnet solid solution model in MELTS and pMELTS is an implementation of the *Berman and Koziol* [1991] model. As adiabat_1ph was being prepared for release, however, it was discovered [Peter Luffi, personal communication 2004] that an error in the printed equations in Berman and Koziol's paper was imported into the MELTS code, and that all versions of MELTS and pMELTS so far released contain this error. Specifically, a factor of 3 in the expression for the logarithm of the activity coefficients is missing, such that the activity coefficients in (p)MELTS are the cubes of the correct values. This makes the garnet model much more non-ideal than it should be, so that the predicted abundance of the grossular component is typically much smaller than experimental values. The published release incorporated, for the first time, the corrected garnet model [Smith and Asimow, 2005].

Note, however, that the MELTS and pMELTS liquid models were calibrated using a few experimental garnet-liquid equilibria, and that these were incorporated into the calibration using the incorrect garnet model. Hence using the model with the new garnet code is not an entirely self-consistent exercise. We believe it will almost always give better garnet mineral chemistry to use the correct model, but other aspects of the calculations (liquid chemistry, melt fraction, modal abundances, etc.) may well be more accurate with the old garnet model. Certainly, to reproduce any previous MELTS or pMELTS calculation, the old garnet model must be used. Therefore, the old garnet model remains the default in adiabat_1ph and the option to use the new garnet model is provided through the environment variable ADIABAT_FIX_GARNET.

Also, small inconsistencies in the annite standard state properties [Berman, 1990] and encoding of the biotite model of *Sack and Ghiorso* [1989] mean that the biotite solid solution in MELTS or pMELTS is always completely dominated by the phlogopite component regards of temperature or composition [Peter Luffi, personal communication 2005]. The biotite model included in adiabat_1ph corrects an error in the coding, 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. Unlike the alternative garnet model, which is optional, the new biotite model is the default in adiabat_1ph.

No biotite-liquid equilibria were included in the MELTS and pMELTS calibrations and so using the new model should not encounter the same problems as for garnet. However, ideal mixing in biotite is almost certainly an oversimplification [Sack and Ghiorso, 1989] and adopting the new model could have undesired effects on the Fe-Mg exchange between biotite and other solid phases such as pyroxenes and garnet [Mark Ghiorso, personal communication, 2005]. The user may prefer to suppress biotite as a phase (see the ‘MELTS file’ section below) to avoid any inconsistencies. For potassium rich compositions, an alternative option in which the biotite solid solution is replaced by a pure-phlogopite phase is provided in ADIABAT_FIX_MICA. When this
environment variable is set for melts like lamprophyres, the behavior will most closely resemble that of other versions of MELTS or pMELTS.

We ask that publication of any results, where garnet and / or biotite were not specifically excluded as phases, indicate which garnet model and /or biotite model was used. We specially emphasize this point because, despite various versions of the interface that have been released, we had until recently been able to maintain complete consistency in the underlying model parameters for all releases of MELTS and pMELTS, such that calculations could be reproduced exactly given only the bulk composition and constraints.

**Bulk composition**

The major and trace element bulk composition must be 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_adiabat.pl 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 adiabat_1ph via a menu option (see ‘Source mixer’). 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 fixed mass of 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. For more details see the ADIABAT_FLUX_MELTING and ADIABAT_ASSIMILATE environment variable entries.

**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. Recently [Thompson, et al., 2007], we have also enabled the isenthalpic mode, 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. Flux melting and assimilation calculations can have non-zero \( P \) and or \( T \) increments in certain circumstances (see ADIABAT_FLUX_MELTING and ADIABAT_ASSIMILATE entries).

Finally, there is a phase diagram orientated mode which 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 pressure 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 an adiabatic
decompression path begins at the solidus, for the given starting pressure. It an also be used like the ‘Find Liquidus’ menu option in the GUI version of MELTS e.g. for fractional crystallization. See menu option 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 in a text file, if the `runadiabat.pl` script is used, or at the menu prompt within `adiabat_1ph`. Once started, the thermodynamic routines will add or drop phases as needed for geochemical equilibrium to be satisfied.

In the GUI version of MELTS or pMELTS, a pure liquid trial solution was used and the algorithm progressively added the solid phases required [see Ghiorso, 1994]. In `adiabat_1ph` the initial guess may be super-liquidus, as before, or sub-solidus [Asimow and Ghiorso, 1998]. In the latter case, the user must specify which solid phases are to be used. A norm calculation is used to find the suitable modal proportions for the given composition, if possible; this routine 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 superliquidus initial guess.

**Trace elements and water partitioning**

Routines for trace element partitioning between melt and solid phases were introduced in pHMELTS [Asimow, et al., 2004]. In `adiabat_1ph` 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].

<table>
<thead>
<tr>
<th>Phase</th>
<th>Reference</th>
<th>Allowed elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clinopyroxene</td>
<td><em>Wood and Blundy [1997]</em></td>
<td>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</td>
</tr>
<tr>
<td></td>
<td><em>Blundy et al. [1995] (for Na)</em></td>
<td></td>
</tr>
<tr>
<td></td>
<td><em>Wood et al. [1999]</em></td>
<td></td>
</tr>
<tr>
<td>Garnet</td>
<td><em>van Westrenen et al [1999]</em></td>
<td>Sc, Y, In, La, Ce, Pr, Nd, Sm, Eu, Gd, Tb, Dy, Ho, Er, Tm, Yb, Lu</td>
</tr>
<tr>
<td>Feldspar</td>
<td><em>Blundy and Wood [1994]</em></td>
<td>Li, Na, K, Rb, Sr, Y, Cs, Ba, La, Ce, Pr, Nd, Sm, Eu, Gd, Tb, Dy, Ho, Er, Tm, Yb, Lu</td>
</tr>
</tbody>
</table>

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.
Previously the solubility of water in olivine in pHMELTS was calculated using the Hirth and Kohlstedt [1996] model; see Asimow, et al. [2004] for more details. 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]. Recent experiments have established that water is incorporated into Fe-bearing olivines more readily than had been assumed [Mosenfelder, et al., 2005]. As a result, pHMELTS now uses the expressions given in Mosenfelder, et al. [2005] by default, although an option to use Hirth and Kohlstedt [1996] is retained for backwards compatibility. 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]. 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 ADIABAT_HK_PXGT_TRACE_H2O and ADIABAT_2X_OPX_TRACE_H2O for 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. For more information on all these options see the ‘Open vs. closed’ section on environment variables and 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. subsolidus or for particular bulk compositions; see ‘General’ 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. However, we have introduced 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 adiabat_1ph.

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 adiabat_1ph 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 integrate_file generated contains the thickness, composition and mean properties of the crust, as well as aggregated liquids at all depths. A stand alone version of the subroutine that performs the integrate calculation, cubicquad2, is also available. If melting is fractional or continuous but performing the pressure integral is inappropriate (e.g. for isobaric heating) then the same routines can be used to output a simple weighted sum of the extracted melts instead.

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.

**Output**

Output includes modes and compositions for all solid and liquid phases, as well as thermodynamic data and density and viscosity estimates. If the appropriate environment variable is set (see ADIABAT_CRASH_FILE below), information is dumped to the named file (the ‘crash_file’) at every step; otherwise output occurs only when requested from the menu.

Importing many different files into spreadsheet programs, such as Excel™, may be time consuming. The Perl script column_pick.pl 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; it may be loaded into Matlab™ using the Import Wizard or with a couple of command line instructions.

**How to cite adiabat_1ph**

The adiabat_1ph front ends are described in:


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, Chemical Mass-Transfer in Magmatic Processes IV. A Revised and Internally Consistent Thermodynamic Model for the Interpolation and


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


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


For calculating variable partition coefficients, please also reference:


A full bibliography is given at the end of this document, which includes references for source compositions and partition coefficients included as part of the programs.

### Setting up adiabat_1ph

**Before you start**

In this documentation we use the following conventions: courier font plain text for commands that must be typed in; square brackets to contain optional parts of the command line (the brackets themselves should not 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 'adiabat_1ph menu prompt' is used to refer to user typed input (as opposed to input read in from a file) while adiabat_1ph is running and 'command line' for things that must be typed at the terminal before or after adiabat_1ph is run.

Windows users should generally substitute 'adiabat_1ph.exe' for 'adiabat_1ph'.

adiabat_1ph or run_adiabat.pl should be called from a terminal program. Linux users will probably be familiar with ‘xterm’ or something similar. In MacOS X the ‘Terminal’ program may be located using ‘Finder’ in Applications/Utilities. Once the program is open, ‘Terminal Help’ offers advice on getting started with the basic Linux / Unix commands needed to navigate around directories, manage files, etc. In Windows, a similar program called ‘Command Prompt’ is in the Start Menu, under Programs/Accessories (or select ‘Run’ and type ‘cmd.exe’); at the Command Prompt type ‘help | more’ to display a list of available commands one screen at a time or
'help command_name' for help on individual commands. Note that 'double clicking' adiabat_1ph/run_adiabat.pl will not generally result in correct behavior.

The Perl program should be included in Linux and MacOS X. If not, or if you are using Windows and have not previously installed Perl, go to http://www.perl.com/download.csp and follow the appropriate link for your operating system. The site http://www.pconline.com/~erc/perlwin.htm may be helpful for Windows users, in addition to the Perl documentation.

**Installation**

Use Stuffit Expander or double click the Disk Image to extract the files for MacOS X. Use the built-in Compressed Folders utility on Windows XP, or WinZip on older Windows versions (see http://www.winzip.com/). To extract the files from the downloaded archive on Linux type:

```bash
tar xvfz linux_adiabat_1ph_*.tar.gz
```

Please join the ‘Adiabat_1ph Users Forum’ at http://macasimow.gps.caltech.edu/~psmith/forum to be notified by e-mail when new versions of the software package are available. Sometimes updates for the adiabat_1ph and / or standalone cubicquad2 program only are made available. If your computer is networked when you run adiabat_1ph and it is more than six hours since the last time it checked the http://www.gps.caltech.edu/~asimow/adiabat server then it will automatically do so and tell you if such updates are available. Do not delete the ‘adiabatdate’ file.

To install the adiabat_1ph update on Windows, replace the adiabat_1ph.exe file in windows_adiabat_1ph_version with the new file. On a Mac (PowerPC) rename the adiabat_1ph_macppc(.txt) executable to adiabat_1ph in OS X’s Finder and put it in the macppc_adiabat_1ph_version folder; for an Intel Mac do the same, replacing ‘ppc’ with ‘i386’. On Linux type the following at the command line:

```bash
mv adiabat_1ph_linux
    path_to_directory/linux_adiabat_1ph_version/adiabat_1ph
```

Update the cubicquad2 executable by following the instructions above replacing each 'adiabat_1ph' with 'cubicquad2'. On Linux and MacOS X it is worth checking that adiabat_1ph and cubicquad2 are executable (see ‘Bugs and Fixes’).

The Perl script run_adiabat.pl must be placed in the same directory as adiabat_1ph. We suggest you add the directory containing adiabat_1ph and run_adiabat.pl to your ‘path’ and then run the Perl script from the directory containing the melts_file (see the adiabat_1ph ‘Interactive Operation’ section).

The ‘path’ is an environment variable that tells the operating system which directories to look for executables in when you type the name of a command. It can be set at the terminal command line. In Windows type:

```bash
set PATH=%PATH%;adiabat_1ph_directory
```

To avoid having to type this each time you open the command prompt go to Control Panel → System → Advanced Tab → Environment Variables → User Variables for your username → New. (Please do not alter the System environment variables unless you are confident you know what you are doing; we will not be responsible for the consequences.) In the pop-up box type Path for the ‘Variable name’ and %PATH%;adiabat_1ph_directory for the ‘Variable Value’.
On Linux or MacOS X use `setenv` or `export` for the (t)csh and bash shells respectively. Type one of the following at the command line, or add a similar line to your ‘.cshrc’ or ‘.bashrc’ file if you do not want to have to repeat this each session:

```
setenv PATH $PATH:/adiabat_1ph_directory
export PATH=$PATH:/adiabat_1ph_directory
```

Alternatively, the following workaround can normally be used: first make a folder called ‘bin’ in your home directory; the ‘bin’ name must be lowercase. At the comment line, navigate to that folder (i.e. ‘/Users/your_username/bin’, on MacOS X, or ‘/home/your_username/bin’, on Linux) and then type following line four times:

```
ln -sf adiabat_1ph_directory/FILE FILE
```

where both instances of ‘FILE’ are replaced in turn by: (1) ‘adiabat_1ph’; (2) ‘run_adiabat.pl’; (3) ‘cubicquad2’ and (4) ‘adiabat_env.dat’. The ‘-sf’ is important as otherwise the software updates will not work correctly. Repeat the process if `adiabat_1ph_directory` changes to a new location.

Contact us (at psmith@gps.caltech.edu) if any of the strategies outlined above do not appear to work. Note `adiabat_1ph_directory` should always be the full path to the directory (e.g. /Users/paula/macppc_adiabat_1ph_2-0 not just macppc_adiabat_1ph_2-0).

**Interactive Operation**

**Synopsis**

`adiabat_1ph`

**Input file formats**

Note that none of the input files are case sensitive but it is important that there are only single spaces (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 should read the ‘Bugs and Fixes’ section for information on making sure that the file formats (specifically line endings) are suitable for the operating system.

**MELTS file**

Input files for the GUI version of MELTS and pMELTS can be used in `adiabat_1ph`, except that some lines (e.g. those beginning ‘Increment’ or ‘Final’) will be ignored as the $P\text{-}T$ path is controlled with environment variables instead. We also introduce the ‘Initial Entropy’ line, ‘Initial Mass’ line and the ‘Initial Trace’ line also which is required if trace element partitioning is switched on; 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. All valid lines are shown in the following example; the ‘Title’ and ‘Initial Composition’ lines are the only compulsory ones (see the note about ‘Initial $P$, $T$ and $fO_2$’). ‘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 printed in any output.
Masses of major oxides (possibly including H2O) are in grams. Temperature is in °C, pressure in bars and entropy in J K\(^{-1}\). The possible settings for the ‘Log fO\(_2\) path’ are FMQ, IW, NNO, HM, NONE and also +3FMQ, +2FMQ, +1FMQ, -1FMQ, -2FMQ and -3FMQ. ‘Log fO\(_2\) delta’ may optionally be used to offset from the chosen buffer in log units (no limit on offset, and need not be an integer); the ‘Log fO\(_2\)’ lines in the example are equivalent to ‘Log fO\(_2\) path: -1FMQ’. If fO\(_2\) buffering is selected and you are using a superliquidus start you may omit the ‘Initial Composition:’ for Fe\(_2\)O\(_3\). For fO\(_2\) buffering with a subsolidus start you must put some of the Fe as Fe\(_2\)O\(_3\) for the initial guess routine to behave correctly but the exact proportions of FeO and Fe\(_2\)O\(_3\) in the melts_file are not important as they will be adjusted by the program. The ‘Initial Mass:’ line is used to scale the system so that total oxides sum to the given number of grams.

```
Title: title_goes_first
Initial Composition: SiO\(_2\) 45.06
Initial Composition: TiO\(_2\) 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: Nd 1.354
Initial Trace: Nd143 0.33917
Initial Trace: Nd144 0.66083
Initial Trace: H2O 200.0

... list all trace elements to be included in the order in which you would like them ...
Initial Mass: 50.0
Initial Temperature: 1100.0
Initial Pressure: 15000.0
Initial Entropy: 260.0
Log fO\(_2\) path: FMQ
Log fO\(_2\) delta: -1.0
Suppress: quartz
... repeat as many times as needed to turn off unwanted phases ...
Fractionate: quartz
Fractionate: feldspar 1
... repeat as many times as needed to fractionate phases selectively...
```

Trace elements 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 the elements exist in the periodic table. There is also 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 FeO* = FeO + 0.9 Fe\(_2\)O\(_3\) in these example files. If fO\(_2\) buffering is not selected it may be desirable to alter the distribution of Fe between FeO and Fe\(_2\)O\(_3\) manually.
As described below, to use the pHMELTS algorithm and treat water as a trace element the environment variable ADIABAT_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 the reference Entropy is set (i.e. 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 adiabat_1ph (in ppm or, if water buffering is chosen, as aH2O), in which case ‘H2O’ will be added to the end of the trace element list if it is not already included.

If ADIABAT_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 ADIABAT_DO_TRACE_H2O but it will not in itself generate an error in adiabat_1ph.

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. See also 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 and it may be help to remove K2O and / or Cr2O3 lines also. Elements that are permitted to have both ‘Initial Composition: oxide’ and ‘Initial Trace: element’ in the same file will behave in a similar manner to H2O with ADIABAT_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 \(^{143}\text{Nd}\) will be 0.459 ppm.

Fractional crystallization of all solid phases (except water) is turned on with the environment variable ADIABAT_FRACTIONATE_SOLIDS. To fractionate only certain phases instead, make sure this environment variable is unset and put ‘Fractionate: phase’ lines in your <i>melts_file</i>. 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 <code>adiabat_1ph</code> 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 <i>melts_file</i> 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. If the exsolved phase tends to join the assemblage again later then the calculation will have to be restarted with the bulk composition and temperature just before the reappearance (the ‘<i>table_file</i>’ function in <code>run_adiabat.pl</code> is potentially useful in this case).

**Enrichment file**

The <i>enrichment_file</i> 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 <i>melts_file</i> (i.e. ‘Initial composition:’ and ‘Initial Trace:’). The trace elements must be in the same order as in the <i>melts_file</i>; otherwise the program with terminate, with an error message. Major element entries are generally optional; see sections on ‘Source mixer’ (menu option 6), ADIABAT_FLUX_MELTING and ADIABAT_ASSIMILATE. There should be a ‘Title:’ line at the start.

Note: those lines that have been read successfully from the <i>melts_file</i> or <i>enrichment_file</i> (and not ignored because they pertain to the GUI versions of MELTS) are printed onscreen. If the line endings in the file do not match your OS and / or there is no newline at the end of the file, the screen output may look a little odd but the program will still have read in the correct values.

**Trace Data file**

The <i>trace_data_file</i> is optional (even when ADIABAT_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 <code>adiabat_1ph</code> 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 <i>trace_data_file</i> listing all the default values is provided (‘default_trace_data.dat’) but is intended as a reference only as these values are already included in the <code>adiabat_1ph</code> code. 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 below.
Set D: olivine Nb 0.005
Set D: feldspar Y 0.03
Constant D: garnet Sc
Variable D: clinopyroxene Sm
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 the environment variable ADIABAT_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 ADIABAT_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 (Eu$^{2+}$ ionic radius from Shannon [1976]). We hope to add redox-dependent Eu partitioning in future.

If ADIABAT_DO_TRACE_H2O is set then mineral-melt partition coefficients for H$_2$O 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 ADIABAT_DO_TRACE_H2O is not set then H$_2$O partition coefficients from the *trace_data_file* will be used but then H$_2$O 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 adiabat_1ph.

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’).

Note that you do not need a trace data file at all if you wish to use only 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).

**Output files**

**Main output file**
The output file will either be the same as the crash file or the file specified by the user during program execution. The values are recorded at each step after equilibrium conditions are achieved and just before any open system behavior (melt extraction, solid fractionation or addition of another increment of assimilant or fluid). A title, taken from the original melts file, is printed first; then six tables are written with one or more lines for each of the calculated equilibria along the path; all tables are space delimited.

- Thermodynamic data; columns are explained in Table 2.
- Liquid compositions in wt% oxides
- The solid phases present and their masses, molar formulas, and wt% compositions
- The masses of all phases present in more compact format than the previous table
- Bulk composition of the solid residue in wt% oxides
- Bulk composition of the system in wt% oxides

**Table 2.**

<table>
<thead>
<tr>
<th>Title</th>
<th>Meaning</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Pressure</td>
<td>bars</td>
</tr>
<tr>
<td>T</td>
<td>Temperature</td>
<td>K (or °C&lt;sup&gt;†&lt;/sup&gt;)</td>
</tr>
<tr>
<td>F</td>
<td>Melt fraction (by mass, relative to the current system mass, prior to any melt extraction)</td>
<td></td>
</tr>
<tr>
<td>phi</td>
<td>Melt fraction (by volume, relative to the current system volume, prior to any melt extraction)</td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>Total entropy of system</td>
<td>J.K&lt;sup&gt;-1&lt;/sup&gt;</td>
</tr>
<tr>
<td>H</td>
<td>Total enthalpy of system</td>
<td>J</td>
</tr>
<tr>
<td>V</td>
<td>Total volume of system</td>
<td>cm&lt;sup&gt;3&lt;/sup&gt;</td>
</tr>
<tr>
<td>dVdP ×10&lt;sup&gt;6&lt;/sup&gt;</td>
<td>Partial derivative of volume with respect to pressure</td>
<td>cm&lt;sup&gt;3&lt;/sup&gt;.bar&lt;sup&gt;-1&lt;/sup&gt;</td>
</tr>
<tr>
<td>dVdT ×10&lt;sup&gt;6&lt;/sup&gt;</td>
<td>Partial derivative of volume with respect to temperature</td>
<td>cm&lt;sup&gt;3&lt;/sup&gt;.K&lt;sup&gt;-1&lt;/sup&gt;</td>
</tr>
<tr>
<td>Cp</td>
<td>Total heat capacity of system</td>
<td>J.K&lt;sup&gt;-1&lt;/sup&gt;</td>
</tr>
</tbody>
</table>

<sup>†</sup> Normally temperature output is in Kelvin but if the ADIABAT_CELSIUS_OUTPUT variable is set then all file and screen output will be in °C. See the relevant environment variable entry for more details.
<table>
<thead>
<tr>
<th>Title</th>
<th>Meaning</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass</td>
<td>Total mass of system</td>
<td>g</td>
</tr>
<tr>
<td>fO2</td>
<td>Oxygen fugacity, relative to assigned buffer or absolute if unbuffered</td>
<td>log 10 units</td>
</tr>
<tr>
<td>Rhol</td>
<td>Density of liquid</td>
<td>g.cm⁻³</td>
</tr>
<tr>
<td>Rhos</td>
<td>Density of bulk solid</td>
<td>g.cm⁻³</td>
</tr>
<tr>
<td>Viscosity</td>
<td>log 10 viscosity of the liquid (Shaw model)</td>
<td>Poise</td>
</tr>
<tr>
<td>aH2O</td>
<td>Activity of water</td>
<td></td>
</tr>
</tbody>
</table>

**Trace element output file**

For each of the calculated equilibria along the path, this outputs:

- Pressure or Temperature, as appropriate for the calculation mode e.g. \( P \) for isentropic.
- \( D \), the bulk partition coefficient for each trace element.
- The composition in ppm or normalized units for each trace element (except \( H_2O \), which is always in ppm) of the Bulk system, Liquid, total Residue and each solid phase in turn.

**Integrated output file**

For the full integration (see ‘Special calculations’ and option 12 below) the first two lines are labels and values for bulk properties of the melting regime if melting continues all the way to the base of the igneous crust. The crust is assumed to have a density of 2.62 g.cm⁻³.

The first seven entries are described in Table 3. 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.

**Table 3.**

<table>
<thead>
<tr>
<th>Title</th>
<th>Meaning</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pc</td>
<td>Thickness of the igneous crust for passive-flow (2-D) integration, in pressure units</td>
<td>bars</td>
</tr>
<tr>
<td>FB</td>
<td>Bulk or mean extent of melting (see Plank et al., [1995]), for 2-D integration with melting stopping at ( Pc ) (by mass, relative to starting mass, for extracted melt only)</td>
<td></td>
</tr>
<tr>
<td>Zc</td>
<td>Thickness of the igneous crust for 2-D integration</td>
<td>km</td>
</tr>
<tr>
<td>(&lt;P&gt;)base</td>
<td>Mean pressure of melt extraction</td>
<td>bars</td>
</tr>
<tr>
<td>Pc1D</td>
<td>Thickness of igneous crust for column-flow (1-D) integration, in pressure units</td>
<td>bars</td>
</tr>
<tr>
<td>Fmax1D</td>
<td>Maximum extent of melting reached at ( Pc1D ) (extracted melt only)</td>
<td></td>
</tr>
<tr>
<td>Zc1D</td>
<td>Thickness of the igneous crust for 1-D integration</td>
<td>km</td>
</tr>
</tbody>
</table>
The next line is a header for all lines to follow (Table 4), and the remaining lines show aggregate properties at each depth in the melting regime, for integration beginning at the solidus (Pressure = \( P_0 \)) and stopping at the depth of interest:

### Table 4.

<table>
<thead>
<tr>
<th>Title (or T)</th>
<th>Meaning</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>P (or T)</td>
<td>Pressure (or Temperature if isobaric, isochoric or DELTAP = 0)</td>
<td>bars (or K(^\dagger))</td>
</tr>
<tr>
<td>F</td>
<td>Aggregated melt fraction at ( P ) (by mass, relative to starting mass, for 1-D integration of extracted melt only)</td>
<td></td>
</tr>
<tr>
<td>P1D</td>
<td>Mean pressure for 1-D integration from ( P_0 ) to ( P )</td>
<td>bars</td>
</tr>
<tr>
<td>P2D</td>
<td>Mean pressure for 2-D integration from ( P_0 ) to ( P )</td>
<td>bars</td>
</tr>
<tr>
<td>iF(dP)</td>
<td>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</td>
<td>bars</td>
</tr>
</tbody>
</table>

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 `cubicquad2` (standalone or internal version) does not perform the pressure integral then the output will be \( P \) (or 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.

**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 thermodynamic data for each phase in the current assemblage, similar to what that printed in its text output.

The first line is a title, followed by a single line of bulk system information for the current conditions with the same format as in Table 2. Output for each stable phase in the assemblage appears next and is as detailed in Table 5. 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 5.

<table>
<thead>
<tr>
<th>Title</th>
<th>Meaning</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Name of the phase</td>
<td></td>
</tr>
<tr>
<td>Mass</td>
<td>Mass of the phase</td>
<td>g</td>
</tr>
<tr>
<td>S</td>
<td>Total entropy of the phase</td>
<td>J,K(^{-1})</td>
</tr>
<tr>
<td>H</td>
<td>Total enthalpy of the phase</td>
<td>J</td>
</tr>
<tr>
<td>Title</td>
<td>Meaning</td>
<td>Units</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>V</td>
<td>Total volume of the phase</td>
<td>cm$^3$</td>
</tr>
<tr>
<td>$dV/dP \times 10^6$</td>
<td>Partial derivative of phase volume with respect to pressure</td>
<td>cm$^3$.bar$^{-1}$</td>
</tr>
<tr>
<td>$dV/dT \times 10^6$</td>
<td>Partial derivative of phase volume with respect to temperature</td>
<td>cm$^3$.K$^{-1}$</td>
</tr>
<tr>
<td>Cp</td>
<td>Total heat capacity of phase</td>
<td>J.K$^{-1}$</td>
</tr>
</tbody>
</table>

The remaining columns alternate between the formula, molar proportion and, for stable phases only, the chemical potential of each end-member in turn. For metastable phases the end-member proportions are the corresponding composition estimate (see Ghiorso [1994]). Only those phases and end-members which fall within the bulk composition space will be printed; so, for example, 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 and then read it in at the start of a new adiabat_1ph session. If the program has just executed (option 4) then the restart file will be for the final conditions after any open system behavior (melt extraction, solid fractionation or addition of another increment of assimilant or fluid).

**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 a superliquidus starting solution will be sought for the current system temperature before trying to equilibrate at 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. However, you can use a subsolidus start if you have previously saved one in a restart file as described below.

First run adiabat_1ph using a *melts_file* for the assimilant composition starting at some much higher temperature. Try an isobaric batch crystallization calculation to the temperature you want the assimilant to be. If the minimization fails before you reach this temperature then run to the last good point; you may need to restart adiabat_1ph and / or set ADIABAT_SKIP_FAILURE. Use menu option 2 to set the temperature to the value you actually want for the assimilant and then save a *restart_file* using menu option 13. When you restart adiabat_1ph for the main assimilation calculation you will be asked whether you want to read in a text or binary file for the assimilant.

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 alright if environment variables from the ‘P-T path control’ (includes ADIABAT_MODE) and ‘Open vs. closed’ sections used when making the *restart_file* do not match the current settings. For assimilation, the $fO_2$ conditions will be the
system ones (e.g. set at the menu or in a melts_file) and any \( fO_2 \) buffer, or lack thereof, in the binary assimilant file will be ignored. See the section on ADIABAT_ASSIMILATE for more details.

**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 adiabat_1ph 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 version 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.

When a text enrichment_file is read in within Source Mixer or during assimilation or flux melting calculations, the bulk composition is adjusted and a new equilibrium assemblage will be found from scratch. Similarly, in most cases when a binary restart_file is used for enrichment, for example during an isenthalpic assimilation calculation, weighted sums of the bulk composition and the relevant reference quantity (enthalpy in the example) are made and a new equilibrium assemblage calculated accordingly. The assumption is that the material being added is far from equilibrium with the current assemblage, due to bulk composition and / or temperature differences. The exception to this general rule of bulk-addition is in Source Mixer when a binary restart_file is read in, as in this case the individual phases are summed. This feature is intended only for adding small amounts of material that is close to equilibrium with the current bulk composition.

A suitable restart_file can be made by fractionating both solids and liquids (turning on both ADIABAT_FRACTIONATE_SOLIDS and ADIABAT_CONTINUOUS) during a melting calculation. First find the appropriate melt fraction using trial-and-error (options 2 and 3; see ‘Menu options’ below). Then set ADIABAT_MINT to the final temperature and rerun so that when option 4 is called a single calculation is followed by fractionation of solids and liquid. It is best to increase ADIABAT_MASSIN, the amount of each solid phase retained, for this stage of the calculation (e.g. to 1 gram). Otherwise minor components, such as K and P, can go below the threshold at which the program considers them to have been exhausted, causing errors. Note that the solids are fractionated before the liquid so that setting ADIABAT_MINF to 0.2 in the above example would give 1 gram each of the liquid and the four solid phases. You will be given the opportunity to scale down the amounts later when reading the restart_file into Source Mixer so that the change in the bulk composition will be trivial.

In general, the program will reliably drop phases that are actually metastable. So, in this example, olivine is immediately dropped in the crystallization run due to its peritectic relationship with orthopyroxene and liquid. Both orthopyroxene and clinopyroxene stay in the assemblage once they
have been added in the ‘seed’ composition. Naturally, it is worth double checking that phases are
indeed dropped as appropriate and making sure that the ‘seed’ assemblage is not too far from
equilibrium with bulk composition to which it is being added.

**Menu options**

The menu commands have the following effects. The first four options will be used in almost all
simple calculations. The next few are for slightly more complicated calculations and the last four
are concerned with output (if there is no crash_file) and quitting.

1. **Read input file to set composition of system**

   Asks for filename of the melts_file, opens it if possible, and sets bulk composition, starting
   pressure, temperature, entropy, log fO\(_2\) path, 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 13) 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 or option 10 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 ADIABAT_DELTAT or ADIABAT_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 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 adiabat_1ph with new P-T limits or increments
   instead; this keeps the batch_file format simple and maximizes backwards compatibility.

   If you are in PTpath mode and have previously executed the program then you will be given the
   opportunity name a new pt.path_file instead. Deliberately or accidentally giving a non-existent
   file name will mean the program reverts to the file set by ADIABAT_PTPATH_FILE instead.

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 ADIABAT_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, 4 or 9 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 Cr$_2$O$_3$ then spinel is needed and K$_2$O 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 (even if ADIABAT_CONTINUOUS or ADIABAT_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).

4. Execute

Begins normal execution at initial conditions, like option 3, and continues along chosen $P$-$T$ path. 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 P-T path 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 H$_2$O buffer and, unless ADIABAT_IMPOSE_FO2 is set, any $fO_2$ buffer will be switched off at this point also.

5. Fix $fO_2$ and / or aH2O

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 the environment variable ADIABAT_IMPOSE_FO2; see the discussion in ‘Open versus closed system behavior’).

The user will then be asked to set aH$_2$O. 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$_2$O is set to 0 then H$_2$O buffering will not be switched on (or will be switched off if previously used). If the ADIABAT_DO_TRACE_H2O environment variable is set, then the total H$_2$O content from the melts_file or Source mixer (menu item 6, see below) will be constrained as the calculation proceeds, so it is not possible to turn on the water buffer and a warning will be given. Like the $fO_2$ buffer, the H$_2$O buffer switches off automatically once the reference entropy, enthalpy or volume is set in isentropic, isenthalpic or isochoric modes. (There is currently no ADIABAT_IMPOSE_FO2 equivalent, although should user-demand arise it could be introduced; see ‘Open versus closed system behavior’).
6. **Source mixer and set H2O**

If water is being treated as a trace element (i.e. if ADIABAT_DO_TRACE_H2O is set) this gives the user an opportunity to reset the current H2O content in ppm. Either way, it then asks for the name of an enrichment_file (text) or restart_file (binary) and opens it if the file exists. To return after setting H2O, without using Source mixer, just give a dummy file name.

If the file is text, then 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 starting 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. Use Source Mixer with this file type for one-off calculation of a new source composition e.g. to model melting of homogenous mixtures differing proportions of peridotite and pyroxenite.

If the file is binary, the amounts and compositions of all phases in the current and restart_file assemblages will be summed individually. The user can enter a scaling factor by which all masses from the restart_file will be multiplied beforehand. 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 explanation of the intended use of combination of menu option and file type.

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 cm\(^3\)). Entering ‘0’ will keep the current reference value, if there is one, or leave it unset. If any calculations have been performed the current total entropy, enthalpy or volume will also be printed for information. 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 then the reference conditions will be calculated for the initial P-T conditions when option 4 is used.

8. **(not used)**

This option will be used for a new feature in an upcoming version of the program. In the meantime it will actually behave like option 14 if selected accidentally.

9. **Turn liquid on / off**

Set whether liquid is allowed or suppressed (overriding any earlier setting from the melts_file). 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 ADIABAT_DELTAP, if appropriate).

10. **Follow mineral isograd or melt fraction contour**

This option starts a routine that seeks the boundary where a phase appears/disappears from the assemblage or where the melt fraction has a particular value, locating by bisection in temperature at each pressure along a path. For the purpose of calculating phase diagrams, set the mode to ‘isothermal’ and specify the pressure increment with the environment variable
ADIABAT_DELTAP. For a single calculation this is not necessary and once the calculation has been performed the user will be given the option to return to the menu and continue execution with option 4. The routine is not totally reliable and some care will be required, for example in the choice of starting conditions, so that a solution can be found.

The reaction may be stable or metastable e.g. see Asimow and Longhi, [2004] for a comparison of stable versus metastable liquidus diagrams. Trace element calculations are only available for stable reactions. For metastable reactions: the user is simply asked which phase to search for and, for a solid, whether the phase is present on the high- or low-temperature side of the boundary. For stable reactions: if no calculations have been performed previously, then the user is prompted for a superliquidus (1) or subsolidus (0) start; see instructions for option 3 for details. In the case of feldspar or clinopyroxene reactions, the user must also tell the program how many copies of the phase to consider. For example for a K-spar boundary in the presence of plagioclase the answer would be ‘2’, as these are both treated as ‘feldspar’. This number is normally one though, especially as adiabat_lph uses pMELTS-like pyroxenes (i.e. clinopyroxene and orthopyroxene are separate phases) even for MELTS calculations. A slower but more rigorous search algorithm can be used for phases that exhibit metastability; this has recently been extended to deal with liquid (e.g. if hysteresis in the melt fraction is observed).

For stable liquid reactions 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 aH$_2$O should only be tried if ADIABAT_DO_TRACE_H2O is unset and the water buffer is off (see option 5). Constant aH$_2$O 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.

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 ADIABAT_MINF or ADIABAT_MINPHI, is the default if ADIABAT_CONTINUOUS is set and may be chosen by giving a negative value for the desired melt fraction. Otherwise the default is 0.0. In order to mimic the ‘Find Liquidus’ menu option in the GUI version of MELTS, choose F or Phi greater than or equal to one and then choose to return to the menu, rather than continuing calculations, when asked.

11. Write output file

Asks for filename, opens it if possible, and writes the standard output information, described in the ‘Main output file’ section.

12. Calculate integrated output file

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. If melting is fractional or continuous but the calculation is isobaric, isochoric or ADIABAT_DELTAP is greater than or equal to zero then the full pressure integral is not performed and a simple weighted sum of the extracted melts is output instead.
The user will be asked to choose Cubic Gaussian Quadrature or a simple rectangle rule for integration; the latter is usually adequate, unless the pressure (or melt fraction) points are unevenly spaced. The major element and, if ADIABAT_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).

13. Read or write restart file
This option allows you to save (in binary format) all the information the program needs to start calculations at the current set of conditions (after any fractionation of solids or liquids, if option 4 has just been run). Asks whether you wish to read or write the file and then for a filename.

Note that if you are reading in restart file(s) with this option, it is crucial that you do this first (i.e. do not read in a melts_file before choosing option 13) and that you use almost exactly the same set of environment variables etc. as was used when the restart_file was generated; check the ‘Restart file’ 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) you will receive error messages if you try to use any other options until a file has been read in.

14. Write melts input 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.

0. Quit
Exits; no files are written. State information is lost.

Environment Variables
Environment variables may be set by hand, using ‘setenv’ or ‘set’ (depending which operating system and / or shell is used). A simpler approach is to run adiabat_1ph from the run_adiabat.pl script. The full range of features are described later in this documentation but briefly the script reads in a list of environment variables from a text file (the command_file) and some environment variables may be set automatically via switches. A default command_file (‘adiabat_env.dat’) is provided with a list of all environment variables with their default values (that is the values they would be assigned within adiabat_1ph if they were not set beforehand).
Note that within adiabat_1ph, 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 ‘adiabat_env.dat’, some lines are commented out with ‘!’ as their default state is to be unset; If run_adiabat.pl detects a ‘!’ in the command file, it ignores the rest of the line. The following environment variables affect adiabat_1ph program execution. Note that they are case sensitive.

**General**

**ADIABAT_VERSION**

*Values:* ‘MELTS’, ‘pMELTS’

*Default:* ‘pMELTS’

Set to choose thermodynamic model. 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.

**ADIABAT_FIX_GARNET**

*Values:* any

*Default:* unset

By default the old (incorrect) garnet model is used so that the thermodynamic models are backwards compatible and consistent with the GUI versions of MELTS and pMELTS. If this environment variable is set then the new, corrected, garnet model is used. This should give better garnet chemistry, but the accuracy of other results (e.g. liquid chemistry, melt fraction, modal abundances) may be affected. See ‘Alternative garnet model’ section for more details.

**ADIABAT_FIX_MICA**

*Values:* any

*Default:* unset

By default the new (revised) biotite model is used but this means that the thermodynamic models may not be backwards compatible or consistent with the GUI versions (opposite to the garnet situation above). If this environment variable is set then the biotite solid solution is replaced by pure phlogopite, which is a viable option if you are studying lamprophyre-like compositions and still wish to maintain reasonable consistency with other versions of MELTS and pMELTS. An alternative approach, for less extreme compositions is to suppress biotite completely (see the ‘MELTS file’ section).

**ADIABAT_ALTERNATIVE_FO2**

*Values:* any

*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
Ghiorsø [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. Overrides ADIABAT_LIQUID_FO2 (next).

**ADIABAT_LIQUID_FO2**

*Values: any*

*Default: unset*

The method of Asimow and Ghiorsø [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. ADIABAT_ALTERNATIVE_FO2 must not be set for this to work.

**ADIABAT_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).

**ADIABAT_FO2_PRESSURE_TERM**

*Values: any*

*Default: unset*

In most versions of MELTS, including Corba MELTS and adiabat_1ph, 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, consistent with Berman [1988], 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 adiabat_1ph 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. Obviously, this environment variable makes no difference for calculations at atmospheric pressure.

**ADIABAT_FLUX_MELTING**

*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’). If used in conjunction with ADIABAT_CONTINUOUS it can simulate flux melting. If trace element calculations are switched on, the enrichment_file must contain the same trace elements as the melts_file. You are asked whether to add major and trace elements or trace elements only, depending on whether you are trying to model fluxing by, say, a metasomatic melt or a hydrous fluid. Normally we recommend running the calculation at constant P, T (i.e. isobaric or isothermal with ADIABAT_DELTAT and / or ADIABAT_DELTAP set to zero). Otherwise, e.g. for calculations along an adiabat, you must choose to mix trace elements only (may include H_2O).

We suggest you set ADIABAT_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. When you select menu option 4 (‘execute’) you should be asked how many iterations you wish to try. After the initial calculation stage, which uses the bulk composition from the melts_file (or restart_file), you will be prompted for the name of the enrichment_file and, if the file exists, the mass of the enrichment composition to be added at each subsequent stage.

**ADIABAT_FLUX_MELTING_PATIENCE**

*Values:* 0 ≤ integer ≤ 100

*Default:* 100

If simulating flux melting, this is the maximum (total) number of iterations that adiabat_lph will run without melting occurring before it gives up and returns to the menu.

**ADIABAT_ASSIMILATE**

*Values:* any

*Default:* unset

This environment variable is similar to ADIABAT_FLUX_MELTING but use of major element mixing is less restricted. It is intended for isobaric / isothermal calculations 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. Solid phases may be fractionated simultaneously.

If the mode is isothermal or ADIABAT_DELTAT is zero and the mode is isobaric or ADIABAT_DELTAP is zero then you will be asked the number of iterations you wish to perform. After the initial calculation the program will request the file type and name and the mass of assimilant to be added at each subsequent stage. As described at length in the ‘Restart file’ section, ADIABAT_ASSIMILATE can read a text or binary file for the assimilant. If running in isothermal mode, or the like, then the P, T and fO_2 buffer from either type of file will be ignored and set to the current values.

For isenthalpic mode: please set ADIABAT_DELTAP to zero, for reasons given in ‘Thermodynamic Path’, and ADIABAT_DELTAT to zero so that the number of iterations can be controlled. In this mode, if you use a text enrichment_file for the assimilant then the program will try to find a thermodynamically equilibrated state, using a superliquidus start. 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.

**P-T path control**

**ADIABAT_MODE**


*Default:* ‘isentropic’

Set the calculation mode. ‘PTpath’ (or ‘PTPath’) reads in \( P \) and \( T \) from a file (see below); in this case ADIABAT_DELTAP is used only to toggle between various output options. For other modes, there are various ways to set initial \( P \), \( T \) and / or reference Entropy, Enthalpy or Volume and the thermodynamic path is set using ADIABAT_DELTAP and / or ADIABAT_DELTAT. The settings are described elsewhere in this documentation.

**ADIABAT_PTPATH_FILE**

*Values:* filename

*Default:* unset

Gives the name of the ptpath_file, which is a simple space delimited text file with one \( P_{\text{value}} T_{\text{value}} \) pair per line.

**ADIABAT_DELTAP**

*Values:* integer or float in bars

*Default:* +1000

Sets pressure increment for isentropic and isothermal modes or if you intend to use the isograd function. This is a signed number; i.e., a positive value steps upwards in pressure, negative steps down.

If ADIABAT_MODE is set to ‘PTpath’ or ‘PTPath’ then ADIABAT_DELTAP may be used to control output options. Set it to zero if Temperature is the more suitable variable for output files. Set it to any negative value if pressures in the ptpath_file decrease monotonically so that thickness, composition and mean properties of the crust for 1-D and 2-D aggregation may be calculated. Set it to any positive value if Pressure is the more suitable output variable but doing the pressure integral in option 12 or the standalone cubicquad2 would be inappropriate.

**ADIABAT_DELTAT**

*Values:* integer or float in °C

*Default:* +10

This sets the temperature increment for isobaric and isochoric modes. This is a signed number; i.e., a positive value steps upwards in \( T \), negative steps down.
ADIABAT_MAXP
Values: integer or float in bars
Default: +30000 if ADIABAT_VERSION = MELTS; +40000 otherwise
Sets the maximum pressure the program will go to.

ADIABAT_MINP
Values: integer or float in bars
Default: +1 if ADIABAT_VERSION = MELTS; +10000 otherwise
Sets the minimum pressure the program will go to.

ADIABAT_MAXT
Values: integer or float in °C
Default: +2000
Sets the maximum temperature the program will go to.

ADIABAT_MINT
Values: integer or float in °C
Default: 0
Sets the minimum temperature the program will go to.

Open vs. closed system

ADIABAT_CONTINUOUS
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 ADIABAT_MINF (or ADIABAT_MINPHI) to 0 for perfect fractional melting. In practice though, the program will run more smoothly if ADIABAT_MINF (or ADIABAT_MINPHI) is slightly greater than 0. See the next four environment variables’ entries for more details.

As of version 1.9 the expression used to calculate the quantity of extracted melt has been tweaked so that the melt fraction is exactly ADIABAT_MINF (or ADIABAT_MINPHI) after extraction. For ‘normal’, e.g. near-fractional, melting calculations where the solid mass is greater than the liquid mass and the amount of melting and ADIABAT_MINF (or ADIABAT_MINPHI) are broadly comparable the difference in results is negligible.

ADIABAT_MINF
Values: 0 ≤ float < 1
Default: 0.005
If ADIABAT_CONTINUOUS 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 ADIABAT_MINF then all the melt will be retained until the next step, otherwise (approximately) F - ADIABAT_MINF will be removed so that the melt fraction is exactly ADIABAT_MINF after extraction (the ‘exactly’ is as of version 1.9 but the difference in values compared to previous versions is negligible). Prior to version 1.7, ‘ADIABAT_MINF’ was actually measured by volume (see below). Liquid and solid densities are similar enough that the difference is small.

**ADIABAT_MINPHI**

*Values: 0 ≤ float < 1*

*Default: unset (or 0.002 if next variable is set)*

If ADIABAT_CONTINUOUS 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 ADIABAT_MINPHI then all the melt will be retained until the next step, otherwise (approximately) phi - ADIABAT_MINPHI will be removed so that the melt fraction, by volume, is exactly ADIABAT_MINPHI after extraction. Overrides ADIABAT_MINF if set by user. Prior to version 1.7, this variable was incorrectly named ‘ADIABAT_MINF’.

**ADIABAT_CONTINUOUS_VOLUME**

*Values: any*

*Default: unset*

This defines a third continuous melting option. Instead of extracting all liquid above a fixed fraction or ratio of melt, 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 ADIABAT_MINPHI variable. This option works in tandem with ADIABAT_MINPHI (or a default value of 0.002 if that is unset) when the solidus is first encountered and overrides it for subsequent extractions. Only use if ADIABAT_MODE is set to ‘isobaric’ or ‘isothermal’. Note that this is not an ‘isochoric’ calculation as far as adiabat_1ph 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.

**ADIABAT_CONTINUOUS_RATIO**

*Values: 0 ≤ float < 1*

*Default: unset*

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. Overrides ADIABAT_MINF or ADIABAT_MINPHI if set.
**ADIABAT_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 certain phases; instead put ‘Fractionate: phase’ lines in your *melts_file* (see the ‘MELTS file’ section above).

**ADIABAT_MASSIN**

*Values: 0 ≤ float < 1*

*Default: 0.001 (or 0.0001 for water)*

Set to the mass in grams of each solid phase retained during fractional crystallization. If the program tends to crash then it is worth trying an increased value. A smaller value can also be used but the program will always round up to a minimum of 10⁻⁶ grams to stabilize the calculation (except for water, which is adjusted independently using ADIABAT_MINW; see next two entries). Once the phase is no longer in the equilibrium assemblage it will be completely exhausted regardless of the value of ADIABAT_MASSIN or ADIABAT_MINW.

**ADIABAT_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 could be extracted during melting rather than just during crystallization.)

**ADIABAT_MINW**

*Values: 0 ≤ float < 1*

*Default: unset*

Set to the proportion of retained water, relative to the total system mass. Works in a similar way, for water, as ADIABAT_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.

**pHMELTS and Trace Elements**

**ADIABAT_DO_TRACE**

*Values: any*

*Default: unset*
Implements attached trace element partitioning function for those elements listed in the `melts_file`.

**ADIABAT_DO_TRACE_H2O**

*Values: any*

*Default: unset*

For the case where water is to be treated as a trace element, this option adds iteration on H$_2$O content as described in Asimow et al. [2004]. For this to work correctly the ADIABAT_DO_TRACE environment variable must also be set, even if no other trace elements are being considered; you will get a warning if it is not.

**ADIABAT_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.

**ADIABAT_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.

**ADIABAT_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 ADIABAT_HK_OL_TRACE_H2O or ADIABAT_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.
**ADIABAT_TRACE_DEFAULT_DPTX**

*Values:* any  
*Default:* unset  

By default all partition coefficients used in trace element calculations are constant. If ADIABAT_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.

**ADIABAT_TRACE_NORMALIZATION**

*Values:* \( 1 \leq \text{integer} \leq 4 \)  
*Default:* unset  

This chooses one of four compositions to normalize trace elements to (if any) or can be left unset for no normalization:

- 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). As described below this environment variable can be set using the \(-s\) command line switch for the `run_adiabat.pl` script and a list of which integers correspond to which compositions can be seen with the \(-h\) switch.

**ADIABAT_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.

**ADIABAT_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.

**ADIABAT_TRACE_OUTPUT_FILE**

*Values:* filename  
*Default: ‘trace.out’*

This gives the name of the file to append trace element information to at each equilibrium step. If the `run_adiabat.pl` script is used it will automatically move any old copy of this file to `filename.old` to help prevent accidental overwriting.

**Others**

**ADIABAT_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.

**ADIABAT_FRACTIONATE_SECOND_LIQUID**

*Values: any  
Default: unset*

When running in ADIABAT_MULTIPLE_LIQUIDS mode, treat all liquids except the first as fractionating phases and remove them from the system after each equilibration.

**ADIABAT_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.

**ADIABAT_FOCUS_FACTOR**

*Values: integer or float  
Default: unset*

When ADIABAT_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 100\(^{th}\) root of 2. If ADIABAT_FOCUS is set without a value in ADIABAT_FOCUS_FACTOR the focusing calculation loop will be ignored.
**Input options**

ADIABAT_V16_BIN_FILE, ADIABAT_V17_BIN_FILE, ADIABAT_V19_BIN_FILE

*Values: any*

*Default: unset*

Unavoidable changes within the program mean that binary *restart files* created with versions 1.6 and earlier of adiabat_1ph will not read into version 1.7. Similarly binary files created with v1.7 will not read into v1.8+ and v1.9 files will not read into v2.0+. To fix this, set one of ADIABAT_V16_BIN_FILE, ADIABAT_V17_BIN_FILE or ADIABAT_V19_BIN_FILE to 'true', depending on the version of your binary file. Once you have read in the file, if you immediately save it again it will be updated to the version 2.0+ format for future use, so unset these environment variables again for subsequent runs.

**Output options**

ADIABAT_CELSIUS_OUTPUT

*Values: any*

*Default: unset*

By default, temperature input to adiabat_1ph is in °C, whereas temperature output is in Kelvin. If this environment variable is set then the temperature output is also in °C. This should cover text file and screen output, including files for trace elements and cubicquad2 (internal or standalone).

ADIABAT_CRASH_FILE

*Values: filename*

*Default: unset*

This gives the name of the file to which output is dumped at every step during execution. This is useful if you need to interrupt the program or the minimization routines crash. If not set, information is only written at the user’s request (option 11), after execution completes. Note that crash file writing is time-consuming; if you are confident that the program will terminate normally and allow you to write an output file then that will be the quicker option. On the other hand, you may lose the whole calculation if it is not being saved in the crash file.

ADIABAT_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 or 9. If this option is set then results from all calculations are saved and output. This provides a simple way to record single calculations (i.e. those from menu option 3) or build up results from multiple iterations or multiple melts files. If this variable is set then the title printed in any output files will always be the one from the first file input. Note that even if the appropriate environment variables are set, such as ADIABAT_FRACTIONATE_SOLIDS
or ADIABAT_CONTINUOUS, no solid or liquid fractionation will occur until menu option 4 is run.

**ADIABAT_SKIP_FAILURE**

*Values: any*

*Default: unset*

Normally failure of the minimization routines means that adiabat_1ph 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 the minimization routines do not recover in the next few iterations then one drawback of ADIABAT_SKIP_FAILURE is that it can lead to infinite loops of failure or, if triggered repeatedly, eventually to a segmentation fault. For this reason, if the minimizations routines fail on two consecutive occasions and one or more of the *PT* limits, such as ADIABAT_MAXP, is reached in each case then ADIABAT_SKIP_FAILURE will abort the calculation. In this way, a judicious choice of the *PT* limits (e.g. setting ADIABAT_MAXP to the initial pressure in isobaric calculations) will mean a clean return to the menu where parameters, such as the choice of *fO₂* buffer, can be adjusted or adiabat_1ph completely restarted.

**ADIABAT_OUTPUT_INTEGRATE_FILE**

*Values: filename*

*Default: unset*

Normally the major oxide integrated output file can be calculated from within adiabat_1ph, but if the standalone program cubicquad2 is being used manually instead (e.g. if adiabat_1ph crashes before menu option 12 can be called) then this environment variable must be set. The resulting file may then be used as input to cubicquad2.

**The Perl script ‘runadiabat.pl’**

The Perl file has a number of options that should make running adiabat_1ph simpler including reading environment variables in from a text file, producing a log of the run conditions and tidying up the program output. Typing ‘runadiabat.pl -h’ will give a brief description of the various command line options; otherwise runadiabat.pl will work through your chosen switches and then start the adiabat_1ph program. (Running with no switches is the same as running adiabat_1ph with default values.) For more information read the comments in the Perl file itself. When the Perl file is run it effectively opens a new shell and sets environment variables for this shell. Therefore, ADIABAT environment variables that are set in your normal shell (e.g. bash or the Windows terminal command line) will not affect program execution.

After execution of adiabat_1ph the *output_file* is processed. The six tables (plus title) mentioned above are split into files named ‘Liq_tbl.txt’, ‘Liq_comp.txt’, ‘Sol_crash.txt’,
‘Phase_props.txt’, ‘Resid_comp.txt’ and ‘Bulk_comp.txt’ making them easier to import into Excel, for example. A tidied up version of the solids file is also written (Solids_comp.txt) in which entries are sorted into separate tables for each phase, comparable to those generated by the GUI version of (p)MELTS, except that solid compositions are listed as oxides rather than end-member proportions and you get six digits rather than two after the decimal point. Any trace element output is tidied and separate tables for D, bulk, Liquid and the various solids are generated in a similar manner. The original trace element file is moved to filename.bak. Optional command line switches for the run_adiabat.pl file are described below.

Please note that, as of version 1.8 the output files, especially the run_adiabat.pl tidied ones have been slightly modified. The changes should be self-explanatory and are designed to make it easier to import the data into Excel™ and MatLab™, particularly as the solid phases are generally not present at all PT points in the adiabat_1ph run (see the ‘column_pick.pl’ section for more details). Version 1.8 also drops the depreciated ‘-n’ switch to run_adiabat.pl as it was not easily compatible with the table_file (‘-t’) option and, in any case, the same effect can be achieved with an ‘Initial Mass:’ line (see the ‘MELTS file’ section).

Synopsis


Typing ‘perl’ at the start and including the path is usually optional. See ‘Bugs and Fixes’ for a list of situations in which it may be necessary to type ‘perl’ etc. and alternative workarounds. Square brackets surround each optional part in the command line for run_adiabat.pl 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.

Input files

Command file

The command_file is the input file for environment variables and other starting conditions. If no name is given the script looks for the default command_file, ‘adiabat_env.dat’, first in the working directory (where the melts_file is) and then in the directory where run_adiabat.pl and adiabat_1ph are. If no command_file is found, the script will stop. As well as the default command_file (‘adiabat_env.dat’), two other sample files (‘frac_xtal.dat’ and ‘isentropic_melt.dat’) are provided and are annotated to show how the command_file may be modified to perform isobaric fractional crystallization and isentropic melting calculations respectively.

The format for most of the file is ADIABAT_ENVIRONMENT_VARIABLE variable_value, as shown in the (isobaric batch melting) example below. Any line that starts with a ‘!’ will be ignored, which provides a quick way to switch environment variables on and off between runs. Initial P, T, entropy and fO2 may be set here using the same format as in the melts_file. The original melts_file will be copied to filename.bak and the Initial P, T, entropy and fO2 lines from the command_file will either replace the corresponding lines from the melts_file or be appended, if no similar lines exist. If the melts_file contains an Initial Entropy line that you wish to ignore put ‘Initial Entropy:
0.0’ in the command_file. Any line-ending issues (see ‘Bugs and Fixes’) should also be fixed in the new melts_file.

```
ADIABAT_VERSION pMelts
ADIABAT_MODE isobaric
ADIABAT_DO_TRACE true
ADIABAT_DELTAT -10
ADIABAT_MINP 20000
... list all environment variable to be set ...
!ADIABAT_CONTINUOUS true
... comment out all environment variables not to be set with ‘!’ ...

Initial Temperature: 1100.0
Initial Entropy: 260.0
Log fO2 path: FMQ
Log fO2 delta: -1.0
... list melts_file style lines to be appended / replaced ...

Suppress: none
 Suppress: olivine
Melts Filename: my_melts_file.melts
Output Filename: my_output_fle.out
Subsolidus phases: olivine clinopyroxene orthopyroxene spinel
```

‘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.

The two ‘Filename:’ lines are equivalent to using the –m and –o switches on the command line respectively. When running in batch mode 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 was running the program by hand. 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.dat’, for simple calculations (no adjusting of parameters) that can be renamed and modified for later use.
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Option 1 to read file</td>
</tr>
<tr>
<td></td>
<td>Name of file</td>
</tr>
<tr>
<td>7</td>
<td>Option 7 to set reference entropy</td>
</tr>
<tr>
<td></td>
<td>Reference Entropy</td>
</tr>
<tr>
<td>3</td>
<td>Option 3 for single calculation</td>
</tr>
<tr>
<td>0</td>
<td>Sub-solidus start</td>
</tr>
<tr>
<td>olivine</td>
<td>list of phases</td>
</tr>
<tr>
<td>clinopyroxene</td>
<td></td>
</tr>
<tr>
<td>orthopyroxene</td>
<td></td>
</tr>
<tr>
<td>spinel</td>
<td></td>
</tr>
<tr>
<td>x</td>
<td>x to end list of phases</td>
</tr>
<tr>
<td>4</td>
<td>Option 4 to execute until max / min P reached</td>
</tr>
<tr>
<td>11</td>
<td>Option 11 to write output_file, if no crash_file</td>
</tr>
<tr>
<td>my_output_file.out</td>
<td>Name of file</td>
</tr>
<tr>
<td>0</td>
<td>Quit</td>
</tr>
</tbody>
</table>

### Switches

**-h**
If this switch is first, run_adiaabat.pl prints the list of all available switches, with a brief explanation of each entry, and then aborts (without running adiabat_lph). Also prints the version number of the script and which versions of adiabat_lph it should be compatible with.

**-p**
*Usage: -p output_path*

The path of the directory to put the output files in (which will be created if it does not exist) relative to the working directory. Note that in Windows any ‘\’ must be replaced by ‘\\’ in output_path. (‘\’ is the escape character in Perl.) For example, the output path could be ‘output/trial1’ in Linux or MacOS X or ‘output\trial1’ in Windows.

**-f**
*Usage: -f command_file*

Input file for environment variables and other starting conditions. If not specified the script will look for a file called ‘adiabat_env.dat’, first in the directory specified by the –p switch (if any) and then in the directory containing adiabat_lph and run_adiaabat.pl.

**-b**
*Usage: -b batch_file*

Run in batch mode. Menu options and filenames etc. that would be typed in during operation of adiabat_lph are instead passed from the user-supplied batch_file by the Perl script.
Automatic mode; this is similar to batch mode, except that the Perl script writes and then uses the simple batch file 'auto_batch.dat' for execution. It assumes that Initial $T$, $P$, $fO_2$ and entropy have already been set (in the melts_file or the command_file). It will use a subsolidus start with phases specified in the command_file if given or a superliquidus start otherwise. It will use the crash_file if specified, or the output_file name given in the command_file.

-m
Usage: -m melts_file
File for initial compositions, etc. If –t is used or if Initial $T$, $P$, $fO_2$ are set in the command_file then run_adiabat.pl needs to know which melts_file to edit before starting adiabat_1ph. The melts_file name can be included in the command_file or it can be specified here, using –m. Otherwise, –m is optional. The melts_file will be copied to filename.bak before any changes are made.

If you are running in automatic mode then the melts_file name will be passed to adiabat_1ph by run_adiabat.pl. Otherwise, if you are not running in batch mode, the file name will be printed just before adiabat_1ph is started so you can copy it when using the menu commands. 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 adiabat_1ph; you still need to use menu option 1.

-t
Usage: -f table_file
This option can be used to automatically generate multiple melts_files from a single melts_file (which must be passed to run_adiabat.pl with the ‘-m’ switch described above), which is used as a template and a space, comma or tab delimited file which contains other bulk compositions and / or initial PT conditions. run_adiabat.pl 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 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_adiabat.pl should deal correctly with ‘wrong’ line-endings so table_files exported from Excel™ can be used without further formatting, even on MacOS X (see ‘Bugs and Fixes’ for more information on issues with line-endings).

Any lines in the melts_file that begin with ‘Initial’ can (optionally) by replaced by a value taken from the table. All other lines will keep their original form and values that do not have ‘Initial’ at the start of the line (e.g.$fO_2$) will have to be changed at the command line or in a batch_file. To be replaced, the column header must be exactly like the appropriate line in the melts_file but stripped of any ‘Initial’, ‘Composition’, ‘Trace’, white space, colon and numerical value (in other words there must be melts_file line something like ‘Initial columnname: value’ or ‘Initial Composition: 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’, ‘FeO’ will match but not ‘FeO*’ and ‘Pressure’ will match whereas ‘P’ runs the risk of being mistaken for phosphorus if that is included as a trace element.
runadiabat.pl 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 header line that gets used or amongst the values) do not contain any white space or commas within them e.g. ‘Experiment’ with values ‘Labell’, ‘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.pl can generally be used as a *table_file*.

If you use automatic mode with a *table_file* then adiabat_1ph will read in and execute each file in turn, starting with the original template *melts_file*. It will use the same superliquidus / subsolidus settings from the *command_file* for each *melts_file* (as described under ‘-a’ above) and the *crash_file* if specified, or the *output_file* name given in the *command_file*; you would need to set ADIABAT_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, if you set one of the *PT* limits, such as ADIABAT_MAXP, so that all initial *PT* points are out-of-limits you can force adiabat_1ph to run a single *PT* point for a variety of bulk compositions.

-o

*Usage:* -o *output_file*

If the ADIABAT_CRASH_FILE environment variable is set then this file will be used when the output is processed by Perl and then –o may be omitted. Otherwise, the name of the *output_file* that will be used in option 11 of adiabat_1ph must either be given in the *command_file*, or it must be passed to Perl on the command line with –o.

If you are running in automatic mode then the *output_file* name will be passed to adiabat_1ph by runadiabat.pl (unless ADIABAT_CRASH_FILE is set, in which case it is not needed). Otherwise, if you are not running in batch mode, the file name will be printed just before adiabat_1ph is started so you can copy it when you use option 11.

-l

*Usage:* -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 *command_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 *command_file* for later runs. If –l is not used this information will be written to ‘logfile.txt’.

-s

*Usage:* -s *integer*

Selects the composition to normalize trace element abundances to (i.e. sets the environment variable ADIABAT_TRACE_NORMALISATION to the integer value given):

- 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].

**Initial P, T, Entropy and fO2**

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

**The standalone ‘cubicquad2’**

**Synopsis**

`cubicquad2`

**Usage**

The standalone version of `cubicquad2` is usually only needed if the calculation crashes before you can use the internal version. Operation is similar to menu option 12, described above, except that the user will be asked to type in an input filename and integers for the number of oxides and trace elements at the `cubicquad2` prompt. The input file for `cubicquad2` is the file specified by the `ADIABAT_OUTPUT_INTEGRATE_FILE` environment variable, which must be set when `adiabat_1ph` is run. Major and trace properties can be output separately, exactly like the internal version. If you prefer a single output file then just give the total number of major and trace elements when prompted for the number of major elements and zero for the number of trace elements. The program prints 'ox1', 'ox2' and 'tr1' etc. in the header lines instead of the real element names. However, it is possible to use option 12 in `adiabat_1ph` between calls to option 3 and option 4; this just outputs correct header lines that can be pasted into the `cubicquad2` output.

**The Perl script ‘column_pick.pl’**

This script allows the user to pick columns from all the various output files to generate a table that will import into Excel and Matable correctly and easily. A copy should be placed in the directory where the output files are. (Alternatively type the path of `column_pick.pl` at the command line or of the desired output file included in the `column_list_file` or both).

The script is simplistic but if used with care it will automatically align and pad columns of data from tables that do not necessarily have the same number of rows. The time taken to get it working properly should be minor compared that saved by importing one, instead of many files.

**Synopsis**

```
[perl -n [-w]] [path]column_pick.pl column_list_file > table_file
```

The ‘`column_list_file`’ 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 (see below). Typing ‘`perl -n`’ at the start and including the path is usually optional. See ‘Bugs and Fixes’ for a list of situations in which it may be necessary to type ‘`perl`’ etc. and alternative workarounds.
The script will not recognize the difference between two columns of the same name. As of version 1.8, all solution phases are output as ‘phase_number’, instead of just liquid, 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’. Case should not matter but the column names must match exactly in all other respects (e.g. ‘pressure’ not ‘P’). Although you can pick out a subset of the columns in any one table, you must still list the columns you want in the same order as they are in the original file. The column order can always be rearranged later once imported in the appropriate software.

The first file must have a line for every P or T, whichever variable is varying (that variable is no longer required to be in the first column). The ‘Variables:’ line is used to read in between 0 and 3 of the first columns of the chosen file. These columns of values will be used to check and align the chosen columns in subsequent files. For files like ‘Liq_tbl.txt’ you can leave this unset. For solid phases: simple calculations (e.g. isobaric) will require ‘Pressure’ or ‘Temperature’, whichever is the one that is varying. Runs in ‘PTpath’ or ‘PTPath’ mode will generally require ‘Pressure Temperature’ or ‘Temperature Pressure’, depending on how ADIABAT_DELTAP is set (see the environment variable section for more details). If the calculation tends to go over the same PT conditions repeatedly e.g. an isobaric-isothermal flux melting calculation then include the ‘mass’ variable as well. The variables must be in the same order as in the next file(s) and you cannot leave one out in the middle i.e. ‘Pressure mass’ or ‘Temperature mass’ will generate an error. ‘Phase_props.txt’ will be the most convenient file to set variables from. 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.

For the ‘Delimiter:’ line, put one of the words ‘space’, ‘comma’ or ‘tab’ or the appropriate character ('	' 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 is it absent will be padded with whichever delimiter was chosen. We have thoroughly tested column_pick.pl and believe it to be reliable; nevertheless it may be worth including the relevant ‘mass’ column to double-check the alignment, as in the example shown (which is for an isobaric run).

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’). In Matlab, use the Import Wizard or try the following command that relies on the file extension ‘.txt’ to detect the file formatting:

```
MyMelts = importdata('table_file.txt')
```

If it works then you will get a structure that includes a cell array containing the column names and a matrix of the numerical data. You can assign columns of the matrix to variable names with something like this (within a loop on ‘k’):

```
eval([MyMelts.colheaders{k}, '=MyMelts.data(:,', int2str(k), ',')]')
```

We have found that comma and tab delimited files are generally more easily imported than space delimited ones, which cause tend to cause confusion with text-only files. If importing the text and
numerical data does not work then upgrade to a more recent version of Matlab™, if available, try running the file through a recent version of Excel™ first or use ‘dlmread’ instead and name the columns by hand. See the help files and documentation in Matlab™ for more information on all these options.

**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 `adiabat_1ph` 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 ADIABAT_CONTINUOUS, then try increasing ADIABAT_MINF slightly (in particular, having it set to 0 can make convergence tricky). Similarly, adjusting ADIABAT_MINW or ADIABAT_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 for a composition that 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 save a `restart_file` (or files) for future use with option 13.

- It is possible that the minimization algorithm in `adiabat_1ph` 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. 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. As of version 1.8, we have 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 than the first attempt. By resetting the $PT$ limits (see menu option 2) and setting ADIABAT_SAVE_ALL you may be able to get the correct results without have to
restart adiabat_lph. 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 ADIABAT_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**

• The following is not really an erratum or a bug but, as it is the most frequent e-mail query, it is included first here. run_adibat.pl only passes the information in your commands_file to adiabat_lph in indirect ways - either by setting environment variables or by making entries in text files. If you are in batch or automatic mode it will redirect the adiabat_lph standard input by executing 'adiabat_lph < batch_file', which passes the melts_file / restart_file name. Otherwise, the user must still prompt adiabat_lph to read in the relevant text or binary files (even when the ‘–m’ run_adibat.pl switch is used); a warning will be given if no files have been input.

Unfortunately, the only short term solution to make run_adibat.pl ‘talk’ directly to adiabat_lph would be to use a program called ‘Expect’. This would mean users would need to download and compile the Perl version of Expect. So that the adiabat_lph package can be portable and require no previous experience with Perl, we hope that you will bear with us by reading in the melts_file with menu option 1 when necessary.

• A common error message is ‘No calculations performed’, which 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 40 kbars for pMELTS and 30 kbars for MELTS calculations.

• The incorporation of Cr$_2$O$_3$ 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 Cr$_2$O$_3$ 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]).

• In Linux / Unix, line endings are marked by the ‘Line Feed’ character (ASCII character 10; ‘\n’). On Macintosh computers, line endings have traditionally been denoted with the ‘Carriage Return’ character (ASCII character 13; ‘\r’; sometimes displays as ‘^M’ or a small box). Windows computers use a ‘Carriage Return’ followed immediately by a ‘Line Feed’. adiabat_lph and run_adibat.pl should tolerate incorrect line endings on most of their input files but there could well be exceptions.

The situation is further complicated in MacOS X as this is a form of Unix and, within the ‘Terminal’ program, it uses the Unix line-endings; so if you edit an input file in Macintosh software (such Microsoft Word or BBEdit) it is possible that adiabat_lph and / or
run_adiabat.pl will not work correctly, even though they are running on the same machine. We suggest you use something like XCode to edit files to help avoid this problem; XCode comes with the MacOS developer tools, which is part of the standard MacOS X distribution, though you may have to expand the package.

To fix line-ending errors we provide a simple Perl script, file_format.pl, which detects what operating system is being used and converts the named text file (e.g. a melts_file or batch_file, with ‘\r’, ‘\n’, or ‘\r\n’ terminators) into the correct format for input into adiabat_1ph or run_adiabat.pl (i.e. Windows or Linux). A copy should be placed in the directory where the relevant files are. (Alternatively type the path of file_format.pl at the command line). To run file_format.pl type:

```perl
[perl –pi[extension]] [path]file_format.pl text_file_name
```

Typing ‘perl -pi’ at the start and including the path is usually optional. See the next bullet point for a list of situations in which it may be necessary to type ‘perl’ etc. and alternative workarounds. The text_file will be overwritten; if the extension is supplied, a backup copy of text_file will be made first with extension added to the name. An extension must be provided in Windows (although it is optional in Linux and MacOS X). Consequently, by default (i.e. if ‘perl –pi’ is omitted) an extension ‘_bak’ is used.

• As mentioned before, when using run_adiabat.pl, column_pick.pl and file_format.pl, typing ‘perl’, ‘perl –n’ and ‘perl –pi’ respectively and including the path to the Perl file is usually optional. However, if Perl is installed somewhere other than /usr/bin on Linux or MacOS X then you will need to type ‘perl’ etc.. Alternatively, you can edit the first line of each Perl file and replace ’/usr/bin’ with the correct location; try typing ‘which perl’ at the terminal command line to find out where Perl is installed.

Similarly typing ‘perl’ etc. will generally only be required on Windows if the ‘.pl’ extension is not registered to Perl (i.e. some other program opens ‘.pl’ files). To register the ‘.pl’ extension to Perl: open Windows Explorer; choose Tools / Folder Options from the menu and select the ‘File Types’ tab; navigate to the ‘PL’ extension and click the ‘Change’ button (you will need to be logged into an account with Administrator privileges to do this); select ‘Perl Command Line Interpreter’ from the list of programs or ‘Browse’ to wherever the ‘perl.exe’ executable is.

It may also be necessary to type ‘perl’ etc. on Windows if the ActivePerl installation is too old. A simpler solution is to update to the latest version of ActivePerl by visiting the http://www.activestate.com/Products/ActivePerl/?_x=1 site.

When typing ‘perl’ etc. the switch –w is not strictly necessary but is recommended as it turns on Perl warnings. The warnings tend to be rather verbose but are less cryptic than the error messages if –w is not used and can be useful indicators for problems like missing files, inappropriate file permissions and incorrect line endings.

• If you get a message like ‘file_name: Permission denied’ in Linux or MacOS X then make sure that adiabat_1ph and / or run_adiabat.pl are executable. Type ‘ls -l
At the terminal command line and see if there is an ‘x’ in the string at the start of the output; if not type `chmod +x file_name` at the terminal command line.

- If `run_adiabat.pl` produces an error message similar to ‘Can’t exec “./adiabat_1ph”’ then make sure that `run_adiabat.pl` and `adiabat_1ph` are in the same directory. Also if you have installed an update binary then make sure you have renamed it appropriately.

- Occasionally `run_adiabat.pl` 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. `adiabat_1ph` did not exit in the expected fashion).

- If you are running `adiabat_1ph` remotely, via `ssh`, you may find that the ‘backspace’ key does not work. If this cannot be fixed and causes inconvenience then try using a `batch_file` for input.

- When you run the `adiabat_1ph` program it will first check the version against the `http://www.gps.caltech.edu/~asimow/adiabat` 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 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:**

pHMELTS, `adiabat_1ph`, and `run_adiabat.pl` 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 should be sent to Antoshechkina (for Perl, output, and front-end issues) or Asimow (for model or algorithm questions) or posted at `http://macasimow.gps.caltech.edu/~psmith/forum` 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 `adiabat_1ph` menu prompt; the `melts_file` and `command_file` you are using; what messages or output you got, if any (for example: on Linux or MacOS X, try copying the screen output to a text file; on Windows, try doing ‘Print Screen’, pasting it into the ‘Paint’ program and saving in JPEG format). 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**


