Equilibrium in CHNOSZ

Jeffrey M. Dick

November 19, 2015

This document defines the concepts, explains the organization of functions, and provides examples of calculating equilibrium in CHNOSZ. It also highlights some applications of the methods (i.e. to reproduce published diagrams) and includes an Appendix on details of the equilibration calculations.

1 Concepts

Species of interest Chemical species for which you want to calculate relative stabilities.

Basis species Species in terms of which you want to write all formation reactions of species of interest.

Formation reactions Stoichiometric chemical reactions showing the mass balance requirements for formation of 1 mole of each species of interest from the basis species.

Chemical affinity Negative of the differential of Gibbs energy of a system with respect to reaction progress. For a given reaction, chemical affinity is the negative of Gibbs energy of reaction; $A = 2.303RT \log(K/Q)$, where K is the equilibrium constant and Q is the activity quotient of species in the reaction (log in this text denotes base-10 logarithms, i.e. `log10` in R).

(1) Reference activity User-defined (usually equal) activities of species of interest.

(1) Reference affinity (A_{ref}) Chemical affinity of formation reaction with a reference activity of the species of interest.

(1) Maximum affinity method Comparison of reference affinities for given balance coefficients in order to calculate stability regions on a predominance diagram.

Balance coefficients ($n_{balance}$) The number of moles of a basis species present in the formation reaction of each of the species of interest. Reactions between any two species of interest then are "balanced" on this basis species. Can be a quantity other than basis species (e.g., balance = 1, or length of amino acid sequence of protein).

Predominance diagram Diagram showing fields of maximal stability (i.e. greatest activity at equilibrium) for species of interest as a function of two variables (aka equal activity diagram).

(2) Starred affinity (A^*) Chemical affinity of formation reaction with unit activity of the species of interest (aka "starved" affinity because the activity of the species of interest drops out of Q).

(2) Total balance activity The sum of activities of this basis species contributed by each of the species of interest. (In Appendix: activity of the immobile or conserved component; a_{ic}.)

(2) Equilibration method Comparison of starred affinities in order to calculate activities of species of interest for given balance coefficients and total balance activity.

Speciation diagram Diagram showing the activities of species of interest, usually as a function of 1 variable (aka activity diagram).

Boltzmann distribution Algorithm used for the equilibration method when the balance coefficients are 1.

Reaction matrix Algorithm used for the equilibration method when the balance coefficients are not all 1.

Normalization Algorithm used for large molecules such as proteins; chemical formulas and affinities are scaled to a similar molecular size (e.g. a single residue; "residue equivalent" in Appendix), activities are calculated using balance = 1, and formulas and activities are rescaled to the original size of the molecule.

Mosaic Calculations of chemical affinities for making diagrams where the speciation of basis species depends on the variables.

The numbered groups above are connected with two distinct approaches to generating diagrams:

1. With the **maximum affinity method** for creating predominance diagrams, the user sets the reference activities of the species of interest; the program compares the reference affinities at these conditions to determine the most stable species (highest activity, i.e. predominant at equilibrium).

2. With the **equilibration method** for creating predominance or activity diagrams, the user explicitly sets the total balance activity or the program takes it from the reference activities of the species. The starred affinities are used to calculate equilibrium activities using one of two techniques (Boltzmann distribution for balance = 1, reaction matrix for balance $\neq$ 1).

The affinities used in these calculations can be calculated using `affinity()`, which works with a single basis set, or with `mosaic()`, which uses multiple basis sets to account for basis species that themselves may change as a function of the variables of interest (e.g. ionization of carbonic acid as a function of pH). This document focuses primarily on the `affinity()` function; for more information on mosaic diagrams see the help page (type `?mosaic` at the R command line).

Step-by-step examples of some of the calculations, particularly the reaction matrix algorithm, are provided in the Appendix. For further description of the equilibration method applied to proteins see Dick and Shock (2013) (also with a derivation of energetic distance from equilibrium using the **starred affinity**).

2 Organization

The function sequences below assume you have already defined the basis species and species of interest using `basis(...)` and `species(...)` (ellipses here and below indicate system-specific input).

Note that if `equilibrate()` or `diagram()` is called without an explicit `balance` argument, the balance coefficients will be taken from the first basis species (in the current basis definition) that is present in all of the species. Depending on the system, this may coincide either with balance = 1 or with balance $\neq$ 1. In the case of `normalize = TRUE` or `as.residue = TRUE`, the balance coefficients (for the purposes of the equilibration step) are temporarily set to 1.

1. Maximum affinity method, balance = 1

    (a) Typical use: simple mineral/aqueous species stability comparisons
    (b) Function sequence:
    ```
    a <- affinity(...)
    diagram(a, balance = 1)
    ```
    (c) Algorithm: $\max \left\{ A_{ref} \right\}$

2. Equilibration method, balance = 1

    (a) Typical use: simple aqueous species activity comparisons
    (b) Function sequence:
    ```
    a <- affinity(...)
    e <- equilibrate(a, balance = 1)
    diagram(e)
    ```

(c) Algorithm: Boltzmann distribution

3. Maximum affinity method, balance $\neq 1$

   (a) Typical use: mineral/aqueous species stability comparisons
   (b) Function sequence:
       ```
       a <- affinity(...)
       diagram(a, balance = ...)
       ```
   (c) Algorithm: $\max\left\{A_{ref}/n_{balance}\right\}$

4. Equilibration method, balance $\neq 1$

   (a) Typical use: aqueous species activity comparisons
   (b) Function sequence:
       ```
       a <- affinity(...)
       e <- equilibrate(a, balance = ...)
       diagram(e)
       ```
   (c) Algorithm: Reaction matrix

5. Maximum affinity method, normalize = TRUE

   (a) Typical use: protein/polymer stability comparisons
   (b) Function sequence:
       ```
       a <- affinity(...)
       diagram(a, normalize = TRUE)
       ```
   (c) Algorithm: $\max\left\{A^*/n_{balance} - \log n_{balance}\right\}$

6. Equilibration method, normalize = TRUE

   (a) Typical use: protein/polymer activity comparisons
   (b) Function sequence:
       ```
       a <- affinity(...)
       e <- equilibrate(a, normalize = TRUE)
       diagram(e)
       ```
   (c) Algorithm: Scale formulas and affinities to residues; Boltzmann distribution (balance = 1); Scale activities to proteins

3 Examples

3.1 Amino acids

Basis species: CO_2, H_2O, NH_3, H_2S, O_2. Species of interest: 20 amino acids. (Only the first few lines of the data frame of amino acid species are shown.)

```
library(CHNOSZ)

## CHNOSZ version 1.0.7 (2015-11-19)
## Please run data(thermo) to create the "thermo" object

data(thermo)

## data(thermo):  attached environment "CHNOSZ"
## thermo$obigt:  1809 aqueous, 3368 total species
```

```
basis("CHNOS")

##     C H N O S ispecies logact state
## CO2 1 0 0 2 0      69     -3    aq
## H2O 0 2 0 1 0       1      0    liq
## NH3 0 3 1 0 0      68     -4    aq
## H2S 0 2 0 0 1      70     -7    aq
## O2  0 0 0 2 0    3095    -80    gas

species(aminoacids(""))[1:5, ]

##   CO2 H2O NH3 H2S    O2 ispecies logact state           name
## 1   3   2   1   0  -3.0     1504     -3    aq        alanine
## 2   3   1   1   1  -2.5     1511     -3    aq       cysteine
## 3   4   2   1   0  -3.0     1509     -3    aq aspartic acid
## 4   5   3   1   0  -4.5     1514     -3    aq glutamic acid
## 5   9   4   1   0 -10.0     1526     -3    aq phenylalanine
```

Code for making the diagrams. Function names refer to the subfigure labels.

```
res <- 200
aa <- aminoacids()

aaA <- function() {
    a <- affinity(O2 = c(-90, -70, res), H2O = c(-20, 10, res))
    diagram(a, balance = 1, names = aa)
}

aaB <- function() {
    a <- affinity(O2 = c(-90, -70, res), H2O = c(-20, 10, res))
    e <- equilibrate(a, balance = 1)
    diagram(e, names = aa)
}

aaC <- function() {
    a <- affinity(O2 = c(-71, -66, res), H2O = c(-8, 4, res))
    diagram(a, balance = "CO2", names = aa)
}

aaD <- function() {
    a <- affinity(O2 = c(-71, -66), H2O = c(-8, 4))
    e <- equilibrate(a, balance = "CO2")
    diagram(e, names = aa)
}

aaE <- function() {
    basis("O2", -66)
    a <- affinity(H2O = c(-8, 4))
    e <- equilibrate(a, balance = "CO2")
    diagram(e, ylim = c(-5, -1), names = aa)
}

aaF <- function() {
    species(1:20, -4)
```

```
    a <- affinity(H2O = c(-8, 4))
    e <- equilibrate(a, balance = "CO2")
    diagram(e, ylim = c(-5, -1), names = aa)
}
```

Note that for the plot we use the 1-letter abbreviations of the amino acids, unlike the full species names (`aminoacids()` is a function in CHNOSZ that returns their names or abbreviations).

```
AA <- aminoacids("")
names(AA) <- aa
AA

##              A              C              D              E                F
##       "alanine"     "cysteine" "aspartic acid" "glutamic acid" "phenylalanine"
##              G              H              I              K                L
##       "glycine"    "histidine"   "isoleucine"      "lysine"       "leucine"
##              M              N              P              Q                R
##    "methionine"   "asparagine"      "proline"     "glutamine"      "arginine"
##              S              T              V              W                Y
##        "serine"    "threonine"        "valine"    "tryptophan"      "tyrosine"
```
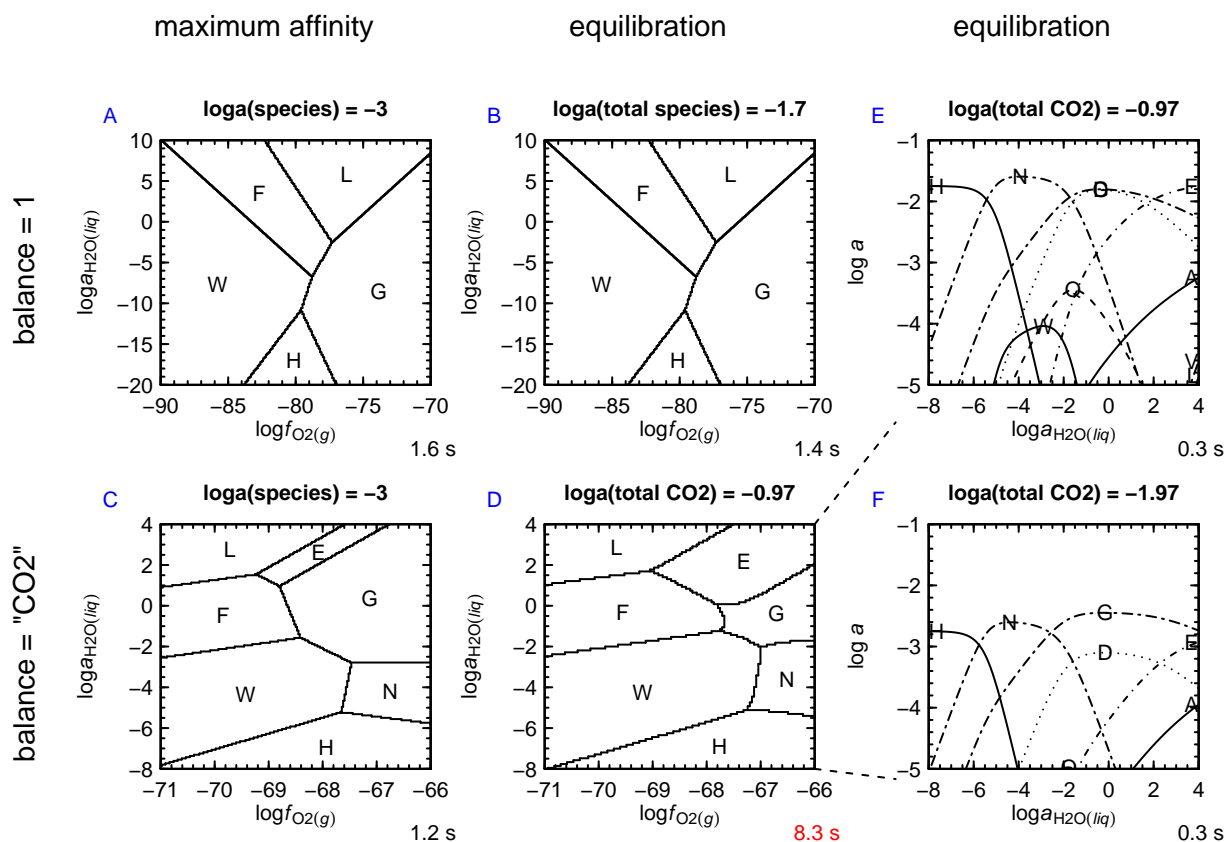
The annotated figure is shown next. The actual code used to set up the plots, add labels, etc. is in the source of this vignette (not shown in the PDF).

Comments on the plots:

- The equal-activity lines in Figures A and B are *identical*. For balance = 1, the maximum affinity method and the equilibration method should produce the same predominance diagrams. (More precisely,

5

because balance = 1, the conditions of equal activity of any species of interest **are independent of** the actual value of that activity.)

- Figures C and D are *different*. For balance $\neq 1$, the maximum affinity method and the equilibration method will generally produce difference predominance diagrams. (Because balance $\neq 1$, the conditions of equal activity of any species of interest **depend on** the actual value of that activity.)

- Both Figures E and F are constructed using the equilibration method, to calculate activities of species as a function of $\log a_{H_2O}$ at $\log f_{O_2} = -66$. Figure E shows the results for the default settings (a_{CO_2} is the sum of activities present in all species, taken from initial species activity of 10^{-3}) and the crossing lines indicating equal activities *are identical to the positions in Figure D* at $\log f_{O_2} = -66$.

- Figure F shows the results for a lower total activity of CO_2. Consequently, the activities of the predominant species decrease from ca. 10^{-2} in Figure E to ca. 10^{-3} in Figure F. Also, the stability region of the smaller glycine has grown at the expense of the neighboring bigger amino acids, so that the crossing lines indicating equal activities in Figure F *are closer to those shown in Figure C* at $\log f_{O_2} = -66$.

- In other words, a lower equal-activity value causes the stability region of the species with the smaller balance coefficient to invade that of the species with the larger balance coefficient.

- Figures A, B, C, and D are all equal activity diagrams, but have different constraints on the activities:

  - Maximum affinity method (Figures A, C): Equal activities of species set to a constant value.
  - Equilibration method (Figures B, D): Equal activities of species determined by overall speciation of the system.

3.2 Proteins

Basis species: CO_2, H_2O, NH_3, H_2S, O_2, H^+. Species of interest: 6 proteins from the set of archaeal and bacterial surface layer proteins considered by Dick (2008).

```
basis("CHNOS+")
organisms <- c("METJA", "HALJP", "METVO", "ACEKI", "GEOSE", "BACLI")
proteins <- c(rep("CSG", 3), rep("SLAP", 3))
species(proteins, organisms)
```

Code for the figures.

```
prA <- function() {
    a <- affinity(O2 = c(-90, -70, res), H2O = c(-20, 0, res))
    e <- equilibrate(a, balance = "length", loga.balance = 0)
    diagram(e, names = organisms)
}

prB <- function() {
    a <- affinity(O2 = c(-90, -70))
    e <- equilibrate(a, balance = "length", loga.balance = 0)
    diagram(e, names = organisms, ylim = c(-5, -1), legend.x = NA)
}

prC <- function() {
    a <- affinity(O2 = c(-90, -70, res), H2O = c(-20, 0, res))
    e <- equilibrate(a, normalize = TRUE, loga.balance = 0)
    diagram(e, names = organisms)
}
```

```
prD <- function() {
    a <- affinity(O2 = c(-90, -70))
    e <- equilibrate(a, normalize = TRUE, loga.balance = 0)
    diagram(e, names = organisms, ylim = c(-5, -1), legend.x = NA)
}

prE <- function() {
    a <- affinity(O2 = c(-90, -70, res), H2O = c(-20, 0, res))
    e <- equilibrate(a, as.residue = TRUE, loga.balance = 0)
    diagram(e, names = organisms)
}

prF <- function() {
    a <- affinity(O2 = c(-90, -70))
    e <- equilibrate(a, as.residue = TRUE, loga.balance = 0)
    diagram(e, names = organisms, ylim = c(-3, 1), legend.x = NA)
}
```

The plots follow. As before, the code used to layout the figure and label the plots is not shown in the PDF.
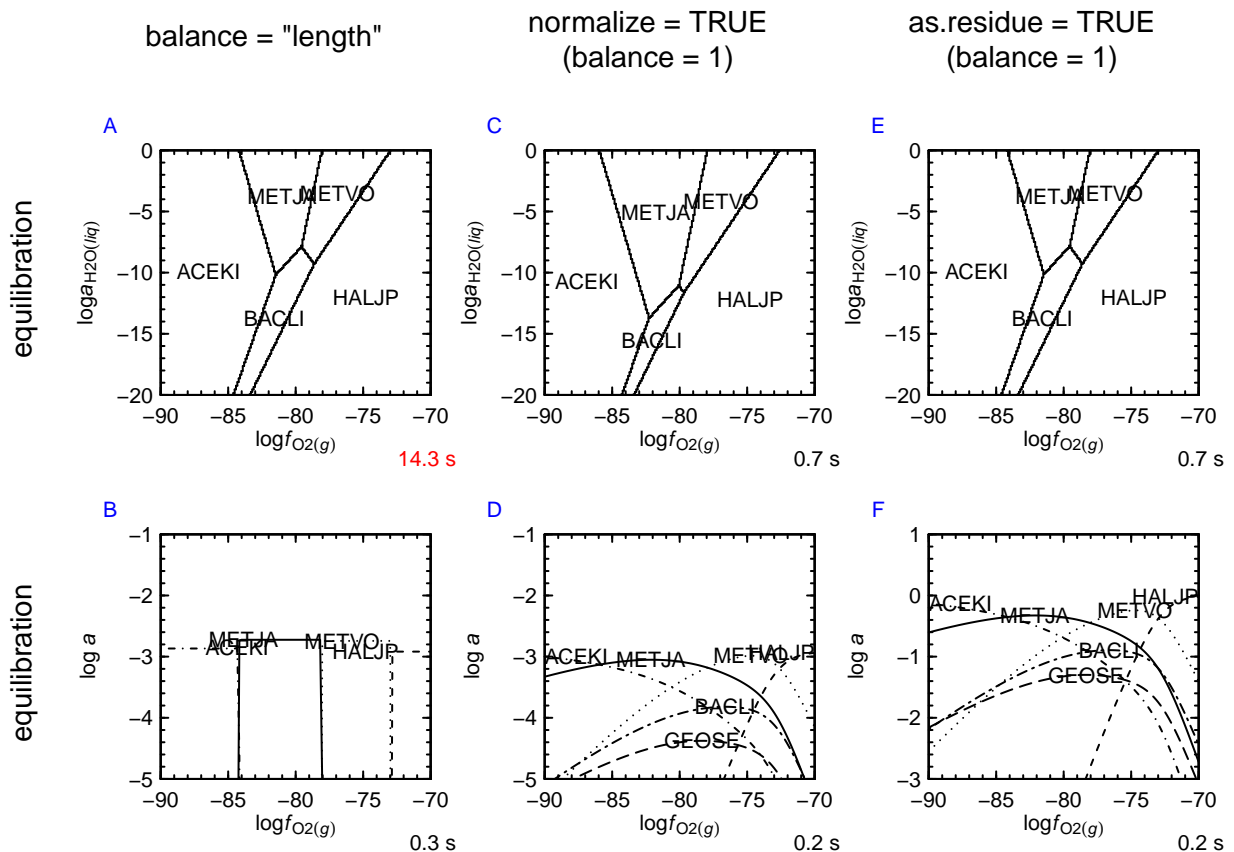

Comments on the plots:

- All of the plots shown are calculated using the equilibration method. The balanced species is amino acid residues (specified by balance = "length" in Figures A and B. In the other figures, normalize = TRUE and as.residue = TRUE internally reset the balance to 1 after scaling the protein formulas to single amino acid residue equivalents). Activity of the balanced species (amino acid residues) is set to 1 (log activity = 0).

7

- Figure B shows that balancing on length produces drastic transitions between activities of the proteins. This either/or type behavior is a consequence of the large sizes of the balancing coefficients, which become exponents on the activities in the expression for Q (or coefficients on the logarithms of activities in $\log Q$).

- Figure D shows that coexistence of proteins with comparable activities can be predicted using `normalize = TRUE`. Here, the protein formulas and affinities are scaled down to their "residue equivalents", then the equilibrium among the residue equivalents is calculated (with balance = 1), and the activities are rescaled to the original proteins. For example, a residue activity of 0 corresponds to 10^{-2} for a 100-residue protein and to 10^{-3} for a 1000-residue protein.

- Figures E and F are like `normalize = TRUE`, except that the rescaling to original protein size is not performed. Note the higher activities of the residue equivalents (Figure F) compared to the proteins (Figure D).

- Compare the equilibration plots above with the maximum affinity plots below. Here the equal activities of the proteins are intentionally set to a very low value: this causes a difference in the plot using balance = "length", but the second and third diagrams remain equivalent to those in Figures C and E above (verified by the `stopifnot` statements; `dA`, `dC` and `dE` refer to the diagrams above). This behavior is consistent with that seen in the amino acid example, where the maximum affinity and equilibration methods give equivalent results for balance = 1 but different results for balance $\neq$ 1.

```
layout(t(matrix(1:3)))
species(1:6, -111)
a <- affinity(O2 = c(-90, -70, res), H2O = c(-20, 0, res))
d1 <- diagram(a, balance = "length", names = organisms, main = "balance = \"length\"")
d2 <- diagram(a, normalize = TRUE, names = organisms, main = "normalize = TRUE")
d3 <- diagram(a, as.residue = TRUE, names = organisms, main = "as.residue = TRUE")
```
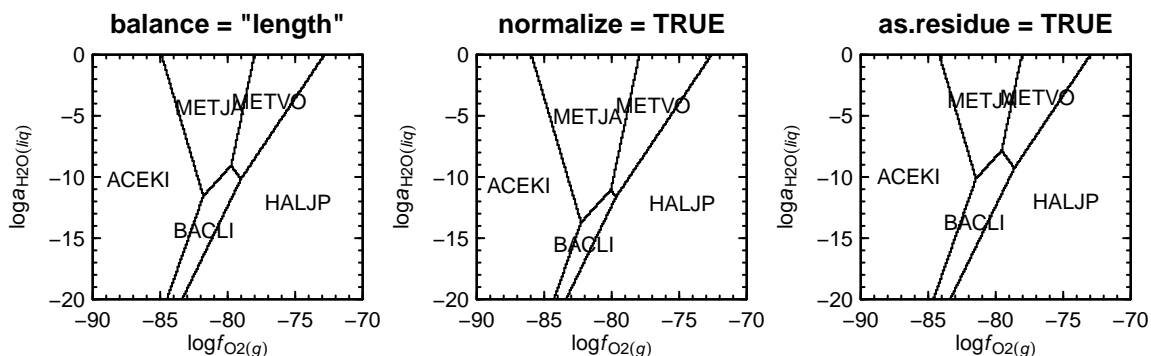


```
stopifnot(!identical(d1$predominant, dA$predominant))
stopifnot(identical(d2$predominant, dC$predominant))
stopifnot(identical(d3$predominant, dE$predominant))
```

With balance = "length", changing the equal activities to *lower values* increases the relative stabilities of the *smaller proteins*, which is why the stability field of the larger protein from BACLI disappears while that of the smaller protein from METJA grows. Because of the drastic activity changes at the stability transitions (see Figure B above), a large change in equal activities (to a minuscule activity = 10^{-111}) is used here to demonstrate this effect, and even then the visual impact on the predominance diagram is subtle. Therefore, naturally occurring relative abundances of proteins are better modeled using the `normalize` or `as.residue` approaches.

4 Applications

Many of the help-page examples and demos in CHNOSZ use these methods to reproduce (or closely emulate) published figures. Below is not a comprehensive list, but just some highlights.

4.1 Maximum affinity method

- The "Aqueous Aluminum" example in `?diagram` shows predominance fields for aqueous species (balance = 1, after Tagirov and Schott, 2001)

- The "Fe-S-O" example in `?diagram` shows stability fields for minerals (balance $\neq$ 1, after Helgeson, 1970).

- Next is an example of using *unequal activities* of species (mineral activity = 1; variable aqueous species activity indicated by the contours (logarithm of activity)) to plot aqueous species – mineral stability boundaries (balance = 1, after Figure 14 of Pourbaix, 1949).

```
basis(c("Cu+2", "H2O", "H+", "e-"))
species(c("Cu+2", "copper", "cuprite", "tenorite"))
for (loga in c(-1, 0, -2, -3)) {
    species("Cu+2", loga)
    a <- affinity(pH = c(1.6, 7.6, 400), Eh = c(-0.2, 1, 400))
    if (loga == -1)
        d <- diagram(a) else d <- diagram(a, add = TRUE, names = NULL)
    iCu <- which(d$predominant == 1, arr.ind = TRUE)
    text(a$vals[[1]][max(iCu[, 1])] - 0.03, a$vals[[2]][min(iCu[, 2])] + 0.2, adj = 1,
        loga)
}
water.lines()
```
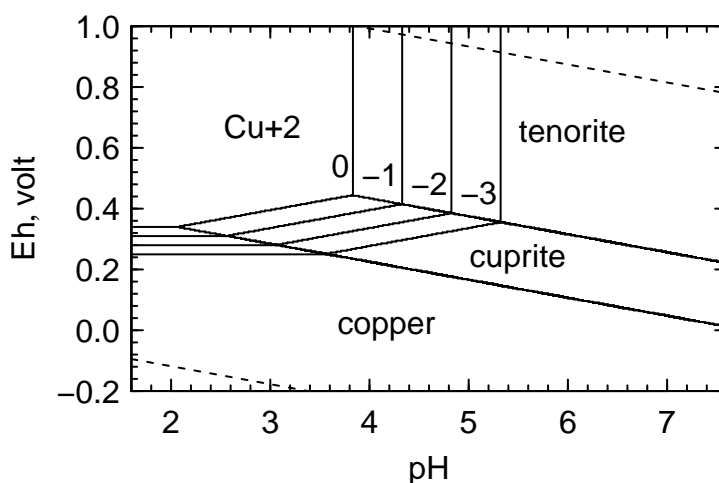

4.2 Equilibration method

- Speciation of reduced and oxidized glutathione, after Schafer and Buettner (2001). Two moles of reduced glutathione (GSH) are oxidized to produce one mole of oxidized glutathione containing a disulfide bond (GSSG), according to

$$2\text{GSH} \rightleftharpoons \text{GSSG} + \text{H}_2$$

9

First, we define a basis set that includes GSH; this becomes the balanced basis species, so we can set its total activity in the call to `equilibrate()`. This total activity is the initial concentration of GSH that will be speciated among GSH and GSSG. If the aqueous species have equal concentrations (or activities), the fraction of GSH that has been oxidized is actually 2/3, because the formation of one mole of GSSG consumes two moles of GSH. That is why the blue lines (fraction of starting GSH that is oxidized) are higher than the black lines (aqueous species distribution). Although the caption to Fig. 3 of Schafer and Buettner (2001) reads "percent GSH that has been oxidized to GSSG", the lines in their figure are closer to the black lines in the figure below.

```
basis(c("GSH", "NH3", "H2S", "H2O", "H+", "e-"))
basis("pH", 7)
species(c("GSH", "GSSG"))
a <- affinity(Eh = c(-0.3, -0.1))
# initial millimoles of GSH
mM <- c(10, 3, 1)
M <- mM * 0.001
for (i in 1:3) {
    e <- equilibrate(a, loga.balance = log10(M[i]))
    diagram(e, alpha = TRUE, lty = c(0, i), add = i != 1, legend.x = NULL, ylim = c(0,
        1), yline = 1.6, lwd = 2, ylab = "aqueous species distribution")
    fGSH <- 1 - (10^e$loga.equil[[1]]/M[i])
    lines(e$vals[[1]], fGSH, col = "blue", lty = i)
}
mtext(side = 2, "fraction of GSH oxidized to GSSG", las = 0, line = 2.6, col = "blue",
    cex = 0.8)
mtext(side = 2, "- - - - - - - - - - - - - - - - - - - -", las = 0, line = 2.1, cex = 0.8)
legend("topleft", lty = 1:3, legend = paste(mM, "mM GSH"))
```
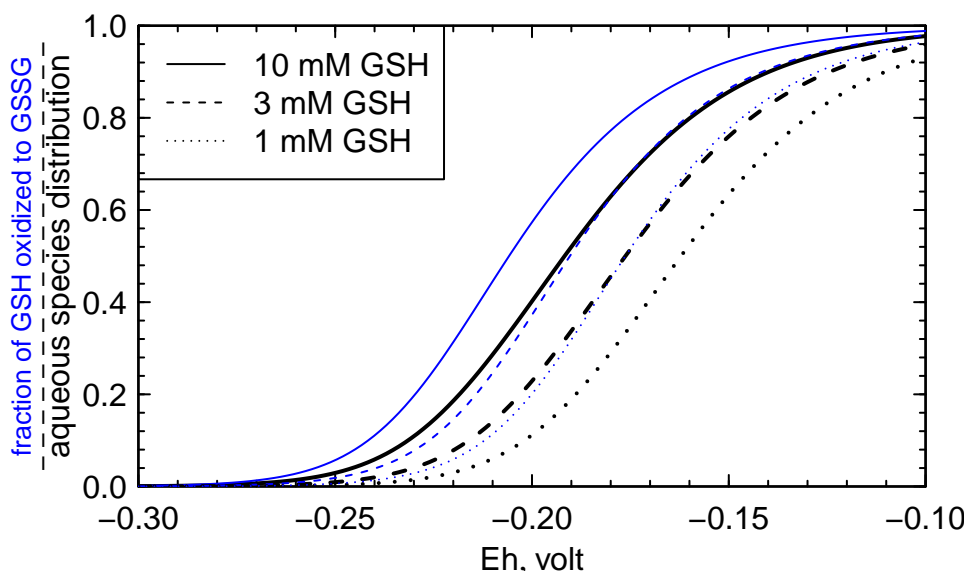

- Protein examples (with `normalize = TRUE`):

  - See "Subcellular proteins" in the vignette *An introduction to CHNOSZ*, "speciation diagram for ER.to.Golgi proteins" in `?read.expr`, and `demo("yeastgfp")`, after Dick (2009).

– Also see the vignette *Hot-spring proteins in CHNOSZ*.

- For systems of many proteins, equilibrium diagrams such as this can become very cluttered, and it may be desirable to use summary statistics to identify trends in the activities or affinities (see `revisit()`).

4.3 Mosaic diagrams

The examples using `mosaic()` shown below are based on the maximum affinity method, but equilibration calculations are also possible (if a suitable example from the literature is found it will be added here; see also the `blend` argument of `mosaic()` to equilibrate the basis species rather than compose the diagram using the predominant basis species).

- See the examples in `?mosaic` and `demo("mosaic")` for calculations of mineral stabilities in the Fe-S-O-H_2O system (after Garrels and Christ, 1965).

- A calculation of copper solubility limits and speciation with aqueous glycine based on Fig. 2b of Aksu and Doyle (2001). We use `mosaic()` to speciate the basis species glycine (activity 10^{-1}) as a function of pH. The stability fields are shown for *unequal activities* of the minerals (unit activity) and aqueous species (10^{-4}). This is essentially a composite of three diagrams, with glycinium, glycine and glycinate in the basis at low, mid and high pH. This demo also modifies the thermodynamic database (with `mod.obigt()`) to use Gibbs energies taken from Aksu and Doyle (2001), and bypasses the default plotting of labels by `diagram()` in order to customize their format and placement.

```
demo("copper", ask = FALSE)
```
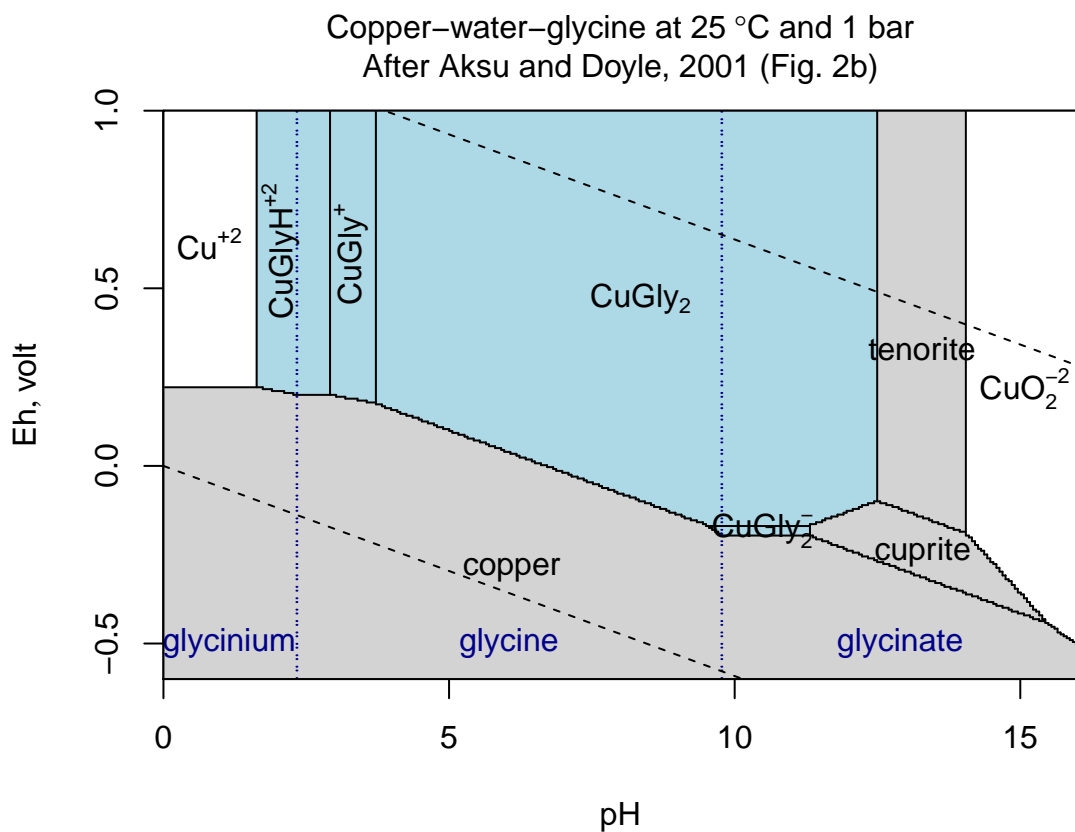
Appendix

Two different methods of calculating the equilibrium activities of species in a system are described below. These are referred to as the *reaction-matrix approach* and the *Boltzmann distribution*. Each method is demonstrated using a specific example that has been described previously (Dick, 2008; Dick and Shock, 2011) (the "CSG" example). The results shows that two approaches are equivalent when the molar formulas are normalized.

A Standard states, the ideal approximation and sources of data

By chemical activity we mean the quantity a_i that appears in the expression

$$\mu_i = \mu_i^\circ + RT \ln a_i \, , \tag{1}$$

where μ_i and μ_i° stand for the chemical potential and the standard chemical potential of the ith species, and R and T represent the gas constant and the temperature in Kelvin (lnhere stands for the natural logarithm). Chemical activity is related to molality (m_i) by

$$a_i = \gamma_i m_i \, , \tag{2}$$

where γ_i stands for the activity coefficient of the ith species. For this discussion, we take $\gamma_i = 1$ for all species, so chemical activity is assumed to be numerically equivalent to molality. Since molality is a measure of concentration, calculating the equilibrium chemical activities can be a theoretical tool to help understand the relative abundances of species, including proteins.

For the CSG examples below, we would like to reproduce exactly the values appearing in publications. Because recent versions of CHNOSZ incorporate data updates for the methionine sidechain group, we should therefore revert to the previous values before proceeding. The `add.obigt()` function does just that, as well as adds other species from the supplemental database provided with CHNOSZ:

```
data(thermo)

## thermo$obigt:  1809 aqueous, 3368 total species

add.obigt()

## add.obigt:  using default file:
## /tmp/RtmpYo79DT/Rinst45b648c18c90/CHNOSZ/extdata/thermo/OBIGT-2.csv
## add.obigt:  read 305 rows; made 84 replacements, 221 additions, units = cal
## add.obigt:  use data(thermo) to restore default database
```

B Reaction-matrix approach

B.1 CSG Example: Whole formulas

Let us calculate the equilibrium activities of two proteins in metastable equilibrium. To do this we start by writing the formation reactions of each protein as

$$stuff_3 \rightleftharpoons \text{CSG\_METVO} \tag{3}$$

and

$$stuff_4 \rightleftharpoons \text{CSG\_METJA} \, . \tag{4}$$

The basis species in the reactions are collectively symbolized by *stuff*; the subscripts simply refer to the reaction number in this document. In these examples, *stuff* consists of CO_2, H_2O, NH_3, O_2, H_2S and H^+ in different molar proportions. To see what *stuff* is, try out these commands in CHNOSZ:

```
basis("CHNOS+")
```

```
##      C H N O S Z ispecies logact state
## CO2 1 0 0 2 0 0       69     -3    aq
## H2O 0 2 0 1 0 0        1      0   liq
## NH3 0 3 1 0 0 0       68     -4    aq
## H2S 0 2 0 0 1 0       70     -7    aq
## O2  0 0 0 2 0 0     3095    -80   gas
## H+  0 1 0 0 0 1        3     -7    aq
```

```
species("CSG", c("METVO", "METJA"))
```

```
##     CO2  H2O NH3 H2S   O2 H+ ispecies logact state       name
## 1 2575 1070 645  11 -2668  0     3590     -3    aq CSG_METVO
## 2 2555 1042 640  14 -2644  0     3591     -3    aq CSG_METJA
```

Although the basis species are defined, the temperature is not yet specified, so it is not immediately possible to calculate the ionization states of the proteins. That is why the coefficient on H^+ is zero in the output above. To see what the computed protein charges are at 25 °C and 1 bar and at pH 7 (specified by the logarithm of activity of H^+ in the basis species), try this:

```
protein.info(species()$name)
```

```
## subcrt:  2 species at 298.15 K and 1 bar (wet)
## subcrt:  18 species at 298.15 K and 1 bar (wet)
## subcrt:  18 species at 298.15 K and 1 bar (wet)
```

```
##     protein length                            formula         G     Z       G.Z       ZC
## 1 CSG_METVO    553 C2575H4040.935N645O884S11-56.065 -24880934 -56.1 -24976763 -0.144
## 2 CSG_METJA    530   C2555H3976.13N640O865S14-55.87 -24236262 -55.9 -24413723 -0.139
```

Note that affinity() is called twice by protein.info(); this so that both charges and standard Gibbs energies of ionization of the proteins can be calculated. The Z values in the table are the charges of the proteins computed using the ionization constants of sidechain and terminal groups, and the G.Z values are the calculated Gibbs energies of formation of the ionized proteins (Dick et al., 2006). The ZC values are the average oxidation states of carbon of the proteins. Let us now calculate the chemical affinities of formation of the ionized proteins:

```
a <- affinity()
```

```
## energy.args:  temperature is 25 C
## energy.args:  pressure is Psat
## subcrt:  8 species at 298.15 K and 1 bar (wet)
## subcrt:  18 species at 298.15 K and 1 bar (wet)
```

```
a$values
```

```
## $`3590`
## [1] 108
##
## $`3591`
## [1] 317
```

Since affinity() returns a list with a lot of information (such as the basis species and species definitions) the last command was written to only print the values part of that list. The values are actually dimensionless, i.e. $A/2.303RT$.

The affinities of the formation reactions above were calculated for a *reference value of activity of the proteins, which is not the equilibrium value*. Those non-equilibrium activities were 10^{-3}. How do we calculate the equilibrium values? Let us write specific statements of the expression for chemical affinity (2.303 is used here to stand for the natural logarithm of 10),

$$A = 2.303RT \log(K/Q), \tag{5}$$

for Reactions 3 and 4 as

$$\begin{aligned} A_3/2.303RT &= \log K_3 - \log Q_3 \\ &= \log K_3 + \log a_{stuff,3} - \log a_{\text{CSG\_METVO}} \\ &= A_3^*/2.303RT - \log a_{\text{CSG\_METVO}} \end{aligned} \tag{6}$$

and

$$\begin{aligned} A_4/2.303RT &= \log K_4 - \log Q_4 \\ &= \log K_4 + \log a_{stuff,4} - \log a_{\text{CSG\_METJA}} \\ &= A_4^*/2.303RT - \log a_{\text{CSG\_METJA}} \,. \end{aligned} \tag{7}$$

The A^* denote the affinities of the formation reactions when the activities of the proteins are unity. I like to call these the "starved" affinities. From the output above it follows that $A_3^*/2.303RT = 104.6774$ and $A_4^*/2.303RT = 314.1877$.

Next we must specify how reactions are balanced in this system: what is conserved during transformations between species (let us call it the immobile component)? For proteins, one possibility is to use the repeating protein backbone group. Let us use n_i to designate the number of residues in the ith protein, which is equal to the number of backbone groups, which is equal to the length of the sequence. If $\gamma_i = 1$ in Eq. (2), the relationship between the activity of the ith protein (a_i) and the activity of the residue equivalent of the ith protein ($a_{residue,i}$) is

$$a_{residue,i} = n_i \times a_i \,. \tag{8}$$

We can use this to write a statement of mass balance:

$$553 \times a_{\text{CSG\_METVO}} + 530 \times a_{\text{CSG\_METJA}} = 1.083 \,. \tag{9}$$

At equilibrium, the affinities of the formation reactions, per conserved quantity (in this case protein backbone groups) are equal. Therefore $A = A_3/553 = A_4/530$ is a condition for equilibrium. Combining this with Eqs. (6) and (7) gives

$$A/2.303RT = \left(104.6774 - \log a_{\text{CSG\_METVO}}\right)/553 \tag{10}$$

and

$$A/2.303RT = \left(314.1877 - \log a_{\text{CSG\_METJA}}\right)/530 \,. \tag{11}$$

Now we have three equations (9–11) with three unknowns. The solution can be displayed in CHNOSZ as follows. Because the balancing coefficients differ from unity, the function called by `equilibrate()` in this case is `equil.reaction()`, which implements the equation-solving strategy described in the next section.

```
e <- equilibrate(a)

## balance:  from protein length
## equilibrate:  n.balance is 553 530
## equilibrate:  loga.balance is 0.0346284566253204
## equilibrate:  using reaction method

e$loga.equil

## [[1]]
## [1] -226
##
## [[2]]
## [1] -2.69
```

Those are the logarithms of the equilibrium activities of the proteins. Combining these values with either Eq. (10) or (11) gives us the same value for affinity of the formation reactions per residue (or per protein backbone group), $A/2.303RT = 0.5978817$. Equilibrium activities that differ by such great magnitudes make it appear that the proteins are very unlikely to coexist in metastable equilibrium. Later we explain the concept of using residue equivalents of the proteins to achieve a different result.

B.2 Implementing the reaction-matrix approach

CHNOSZ implements a method for solving the system of equations that relies on a difference function for the activity of the immobile component. The steps to obtain this difference function are:

1. Set the total activity of the immobile (conserved) component (aka total balance activity) as a_{ic} (e.g., the 1.083 in Eqn. 9).

2. Write a function for the logarithm of activity of each of the species of interest: $A = (A_i^* - 2.303RT \log a_i)/n_{ic,i}$, where $n_{ic,i}$ stands for the number of moles of the immobile component that react in the formation of one mole of the ith species. (e.g., for systems of proteins where the backbone group is conserved, $n_{ic,i}$ is the same as n_i in Eq. 8). Calculate values for each of the A_i^*. Metastable equilibrium is implied by the equality of A in all of the equations.

3. Write a function for the total activity of the immobile component: $a_{ic}' = \sum n_{ic,i} a_i$.

4. The difference function is now $\delta a_{ic} = a_{ic}' - a_{ic}$.

Now all we have to do is find the value of A where $\delta a_{ic} = 0$. This is achieved in the code by first looking for a range of values of A where at one end $\delta a_{ic} < 0$ and at the other end $\delta a_{ic} > 0$, then using the `uniroot()` function that is part of R to find the solution.

Even if values of δa_{ic} on either side of zero can be located, the algorithm does not guarantee an accurate solution and may give a warning about poor convergence if a certain tolerance is not reached.

B.3 CSG Example: normalized formulas (residue equivalents)

Let us consider the formation reactions of the normalized formulas (residue equivalents) of proteins, for example

$$stuff_{12} \rightleftharpoons \text{CSG\_METVO(residue)} \tag{12}$$

and

$$stuff_{13} \rightleftharpoons \text{CSG\_METJA(residue)}. \tag{13}$$

The formulas of the residue equivalents are those of the proteins divided by the number of residues in each protein. The `protein.basis()` function shows the coefficients on the basis species in these reactions:

```
protein.basis(species()$name, normalize = TRUE)

## subcrt:  18 species at 298.15 K and 1 bar (wet)

##        CO2  H2O  NH3   H2S    O2     H+
## [1,] 4.66 1.93 1.17 0.0199 -4.82 -0.101
## [2,] 4.82 1.97 1.21 0.0264 -4.99 -0.105
```

Let us denote by A_{12} and A_{13} the chemical affinities of Reactions 12 and 13. We can write

$$A_{12}/2.303RT = \log K_{12} + \log a_{stuff,12} - \log a_{\text{CSG\_METVO(residue)}} \tag{14}$$

and

$$A_{13}/2.303RT = \log K_{13} + \log a_{stuff,13} - \log a_{\text{CSG\_METJA(residue)}}. \tag{15}$$

For metastable equilibrium we have $A_{12}/1 = A_{13}/1$. The 1's in the denominators are there as a reminder that we are still conserving residues, and that each reaction now is written for the formation of a single residue equivalent. So, let us write A for $A_{12} = A_{13}$ and also define $A_{12}^* = A_{12} + 2.303RT \log a_{\text{CSG\_METVO(residue)}}$ and $A_{13}^* = A_{13} + 2.303RT \log a_{\text{CSG\_METJA(residue)}}$. At the same temperature, pressure and activities of basis species and proteins as shown in the previous section, we can write $A_{12}^* = A_3^*/553 = 2.303RT \times 0.1892901$ and $A_{13}^* = A_4^*/530 = 2.303RT \times 0.5928069$ to give

$$A/2.303RT = 0.1892901 - \log a_{\text{CSG\_METVO(residue)}} \tag{16}$$

and

$$A/2.303RT = 0.5928069 - \log a_{\text{CSG\_METJA(residue)}}, \tag{17}$$

which are equivalent to Equations 12 and 13 in the paper (Dick, 2008) but with more decimal places shown. A third equation arises from the conservation of amino acid residues:

$$a_{\text{CSG\_METVO(residue)}} + a_{\text{CSG\_METJA(residue)}} = 1.083. \tag{18}$$

The solution to these equations is $a_{\text{CSG\_METVO(residue)}} = 0.3065982$, $a_{\text{CSG\_METJA(residue)}} = 0.7764018$ and $A/2.303RT = 0.7027204$.

The corresponding logarithms of activities of the proteins are $\log (0.307/553) = -3.256$ and $\log (0.776/530) = -2.834$. These activities of the proteins are much closer to each other than those calculated using formation reactions for whole protein formulas, so this result seems more compatible with the actual coexistence of proteins in nature.

The approach just described is not actually used by `equilibrate(..., normalize = TRUE)`. Instead, because balance = 1, the Boltzmann distribution, which is faster, can be used.

C Boltzmann distribution

C.1 CSG Example: Normalized formulas

An expression for Boltzmann distribution, relating equilibrium activities of species to the affinities of their formation reactions, can be written as (using the same definitions of the symbols above)

$$\frac{a_i}{\sum a_i} = \frac{e^{A_i^*/RT}}{\sum e^{A_i^*/RT}}. \tag{19}$$

Using this equation, we can very quickly (without setting up a system of equations) calculate the equilibrium activities of proteins using their residue equivalents. Above, we saw $A_{12}^*/2.303RT = 0.1892901$ and $A_{13}^*/2.303RT = 0.5928069$. Multiplying by $\ln 10 = 2.302585$ gives $A_{12}^*/RT = 0.4358565$ and $A_{13}^*/RT = 1.364988$. We then have $e^{A_{12}^*/RT} = 1.546287$ and $e^{A_{13}/RT} = 3.915678$. This gives us $\sum e^{A_i^*/RT} = 5.461965$, $a_{12}/\sum a_i = 0.2831009$ and $a_{13}/\sum a_i = 0.7168991$. Since $\sum a_i = 1.083$, we arrive at $a_{12} = 0.3065982$ and $a_{13} = 0.7764018$, the same result as above.

D Notes on implementation

D.1 CSG example: another look

All the tedium of writing reactions, calculating affinities, etc., above does help to understand the application of the reaction-matrix and Boltzmann distribution methods to protein equilibrium calculations. But can we automate the step-by-step calculation for any system, including more than two proteins? And can we be sure that higher-level functions in CHNOSZ, particularly `equilibrate()`, match the output of the step-by-step calculations? Now we can, with the `protein.equil()` function introduced in version 0.9-9. Below is its output when configured for CSG example we have been discussing.

```
protein <- iprotein(c("CSG_METVO", "CSG_METJA"))
basis("CHNOS+")
swap.basis("O2", "H2")
protein.equil(protein, loga.protein = -3)

## protein.equil:  temperature from argument is 25 degrees C
## protein.equil:  pH from thermo$basis is 7
## protein.equil:  [Met] is from reference DLH06
## protein.equil [1]:  first protein is CSG_METVO with length 553
## protein.equil [1]:  reaction to form nonionized protein from basis species has G0(cal/mol) of -47580000
## protein.equil [1]:  ionization reaction of protein has G0(cal/mol) of -95830
## protein.equil [1]:  per residue, reaction to form ionized protein from basis species has G0/RT of
-145.5
## protein.equil [1]:  per residue, logQstar is 63.01
## protein.equil [1]:  per residue, Astar/RT = -G0/RT - 2.303logQstar is 0.4359
## check it!    per residue, Astar/RT calculated using affinity() is 0.4359
## protein.equil [all]:  lengths of all proteins are 553 530
## protein.equil [all]:  Astar/RT of all residue equivalents are 0.4359 1.365
## protein.equil [all]:  sum of exp(Astar/RT) of all residue equivalents is 5.462
## protein.equil [all]:  equilibrium degrees of formation (alphas) of residue equivalents are 0.2831
0.7169
## check it!    alphas of residue equivalents from equilibrate() are 0.2831 0.7169
## protein.equil [all]:  for activity of proteins equal to 10^-3, total activity of residues is 10^0.03463
## protein.equil [all]:  log10 equilibrium activities of residue equivalents are -0.5134 -0.1099
## protein.equil [all]:  log10 equilibrium activities of proteins are -3.256 -2.834
## check it!    log10 eq'm activities of proteins from equilibrate() are -3.256 -2.834
```

The function checks ("check it!") against the step-by-step calculations the values of A^* calculated using `affinity()`, and the equilibrium activities of the proteins calculated using `equilibrate()`. (Note that Astar/RT in the second line after the first "check it!" can be multiplied by ln 10 to get the values shown above in Eqs. 16 and 17.)

D.2   Visualizing the effects of normalization

A comparison of equilibrium calculations that do and do not use normalized formulas for proteins was presented by Dick (2008). A diagram like Figure 5 in that paper is shown below.

```
organisms <- c("METSC", "METJA", "METFE", "HALJP", "METVO", "METBU", "ACEKI", "GEOSE",
    "BACLI", "AERSA")
proteins <- c(rep("CSG", 6), rep("SLAP", 4))
basis("CHNOS+")
species(proteins, organisms)
a <- affinity(O2 = c(-100, -65))
layout(matrix(1:2), heights = c(1, 2))
e <- equilibrate(a)
diagram(e, ylim = c(-2.8, -1.6), legend.x = NA, names = organisms)
water.lines(xaxis = "O2")
title(main = "Equilibrium activities of proteins, normalize = FALSE")
e <- equilibrate(a, normalize = TRUE)
diagram(e, ylim = c(-4, -2), legend.x = NA, names = organisms)
water.lines(xaxis = "O2")
title(main = "Equilibrium activities of proteins, normalize = TRUE")
```
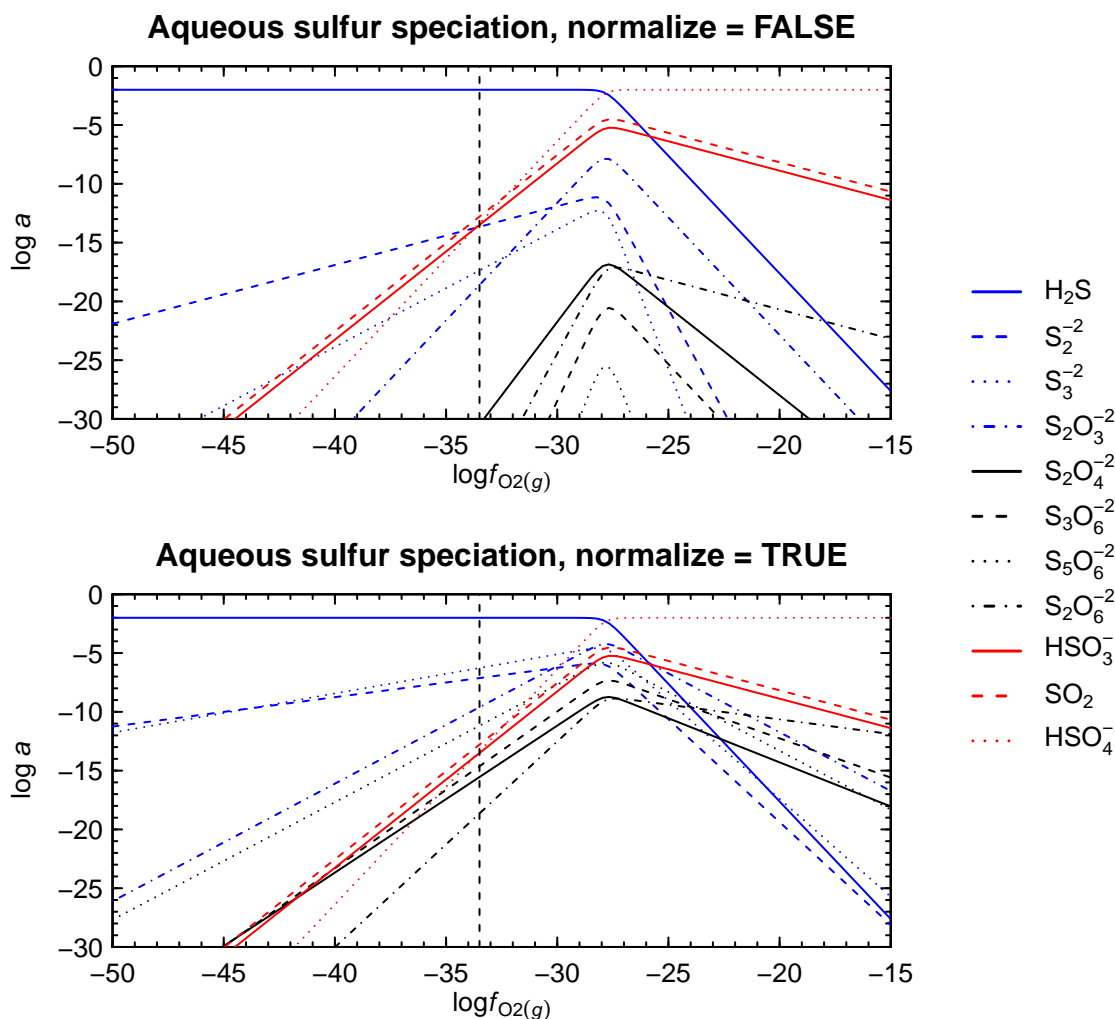
17

Equilibrium activities of proteins, normalize = FALSE

Equilibrium activities of proteins, normalize = TRUE
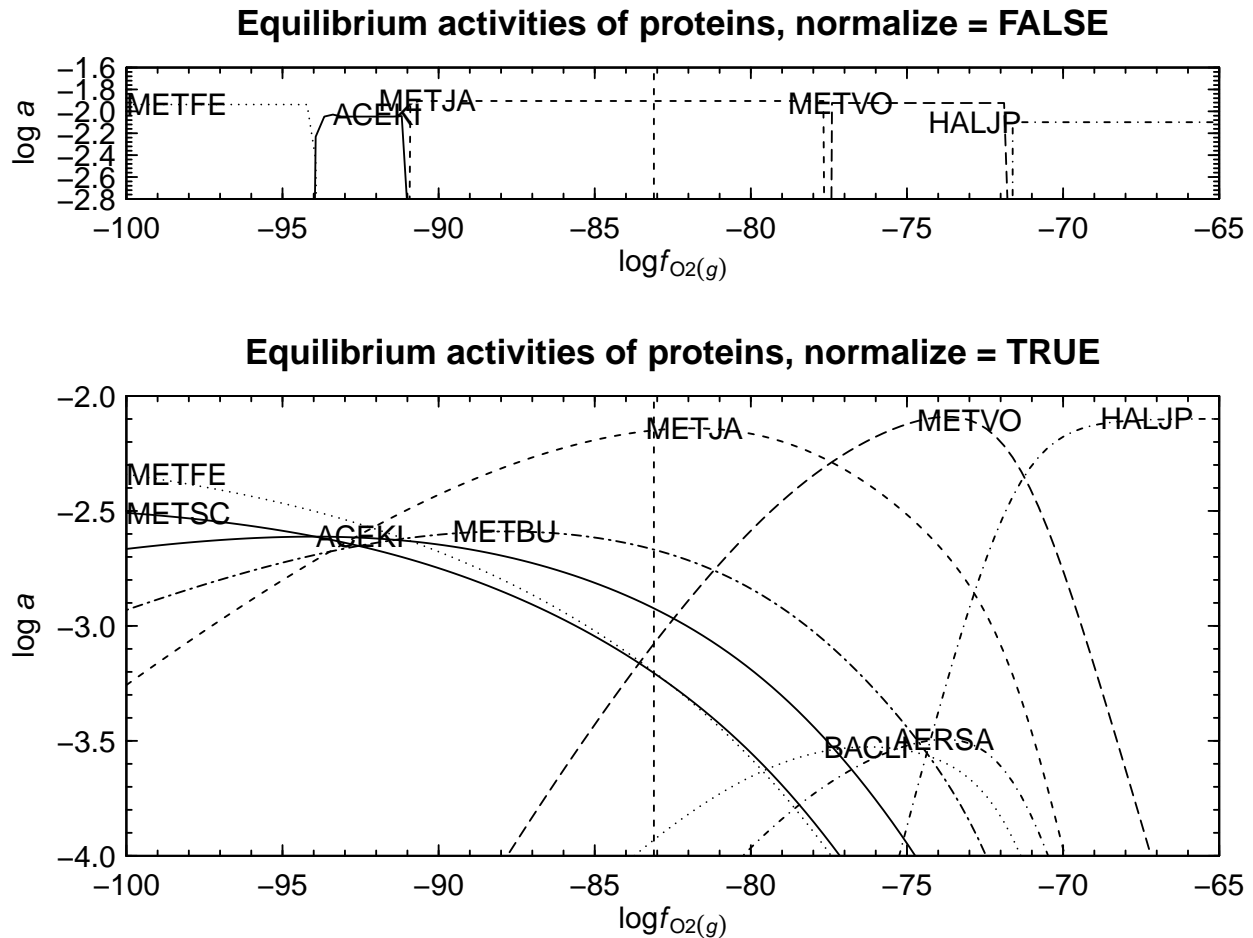
Although it is well below the stability limit of H_2O (vertical dashed line), there is an interesting convergence of the activities of some proteins at low $\log f_{O_2}$, due most likely to compositional similarity of the amino acid sequences.

The reaction-matrix approach can also be applied to systems having conservation coefficients that differ from unity, such as many mineral and inorganic systems, where the immobile component has different molar coefficients in the formulas. For example, consider a system like one described by Seewald (1996):

```r
basis("CHNOS+")
basis("pH", 5)
species(c("H2S", "S2-2", "S3-2", "S2O3-2", "S2O4-2", "S3O6-2", "S5O6-2", "S2O6-2", "HSO3-",
    "SO2", "HSO4-"))
a <- affinity(O2 = c(-50, -15), T = 325, P = 350)
layout(matrix(c(1, 2, 3, 3), nrow = 2), widths = c(4, 1))
col <- rep(c("blue", "black", "red"), each = 4)
lty <- 1:4
for (normalize in c(FALSE, TRUE)) {
    e <- equilibrate(a, loga.balance = -2, normalize = normalize)
    diagram(e, ylim = c(-30, 0), legend.x = NULL, col = col, lty = lty)
    water.lines(xaxis = "O2", T = convert(325, "K"))
    title(main = paste("Aqueous sulfur speciation, normalize =", normalize))
}
par(mar = c(0, 0, 0, 0))
plot.new()
```

18

```
leg <- lapply(species()$name, expr.species)
legend("center", lty = lty, col = col, lwd = 1.5, bty = "n", legend = as.expression(leg),
    y.intersp = 1.3)
```


Aqueous sulfur speciation, normalize = FALSE

Aqueous sulfur speciation, normalize = TRUE

Legend: H_2S, S_2^{-2}, S_3^{-2}, $S_2O_3^{-2}$, $S_2O_4^{-2}$, $S_3O_6^{-2}$, $S_5O_6^{-2}$, $S_2O_6^{-2}$, HSO_3^-, SO_2, HSO_4^-

The first diagram is quantitatively similar to the one shown by Seewald (1996), with the addition of color and the water stability line. If we use the normalized formulas (divide whole formulas by moles of H_2S in their formation reactions), the range of activities of species is smaller at any given log $f_{O_{2(g)}}$, as shown in the second diagram. Although `normalize = TRUE` was implemented to study the relative stabilities of proteins, it might also be relevant to other systems where the molecules are in various stages of polymerization.

Document history

- 2009-11-29 Initial version containing CSG example (title: Calculating relative abundances of proteins)

- 2012-09-30 Renamed to "Equilibrium in CHNOSZ". Remove activity comparisons, add maximum affinity method.

- 2015-11-08 Add sections on concepts, organization, examples and applications; move most material from the previous version of the document to the Appendix; now uses the **knitr** vignette engine.

References

Serdar Aksu and Fiona M. Doyle. Electrochemistry of copper in aqueous glycine solutions. *Journal of the Electrochemical Society*, 148(1):B51–B57, 2001. doi: 10.1149/1.1344532.

Jeffrey M. Dick. Calculation of the relative metastabilities of proteins using the CHNOSZ software package. *Geochemical Transactions*, 9:10, 2008. doi: 10.1186/1467-4866-9-10.

Jeffrey M. Dick. Calculation of the relative metastabilities of proteins in subcellular compartments of *Saccharomyces cerevisiae*. *BMC Systems Biology*, 3:75, 2009. doi: 10.1186/1752-0509-3-75.

Jeffrey M. Dick and Everett L. Shock. Calculation of the relative chemical stabilities of proteins as a function of temperature and redox chemistry in a hot spring. *PLoS ONE*, 6(8):e22782, 2011. doi: 10.1371/journal.pone.0022782.

Jeffrey M. Dick and Everett L. Shock. A metastable equilibrium model for the relative abundances of microbial phyla in a hot spring. *PLoS ONE*, 8(9):e72395, 2013. doi: 10.1371/journal.pone.0072395.

Jeffrey M. Dick, Douglas E. LaRowe, and Harold C. Helgeson. Temperature, pressure, and electrochemical constraints on protein speciation: Group additivity calculation of the standard molal thermodynamic properties of ionized unfolded proteins. *Biogeosciences*, 3(3):311–336, 2006. doi: 10.5194/bg-3-311-2006.

Robert M. Garrels and Charles L. Christ. *Solutions, Minerals, and Equilibria*. Harper & Row, New York, 1965. URL http://www.worldcat.org/oclc/517586.

Harold C. Helgeson. A chemical and thermodynamic model of ore deposition in hydrothermal systems. In Benjamin A. Morgan, editor, *Fiftieth Anniversary Symposia*, volume 3 of *Mineralogical Society of America, Special Paper*, pages 155–186. Mineralogical Society of America, 1970. URL http://www.worldcat.org/oclc/583263.

Marcel J. N. Pourbaix. *Thermodynamics of Dilute Aqueous Solutions*. Edward Arnold & Co., London, 1949. URL http://www.worldcat.org/oclc/1356445.

Freya Q. Schafer and Garry R. Buettner. Redox environment of the cell as viewed through the redox state of the glutathione disulfide/glutathione couple. *Free Radical Biology and Medicine*, 30(11):1191–1212, 2001. doi: 10.1016/S0891-5849(01)00480-4.

Jeffrey S. Seewald. Mineral redox buffers and the stability of organic compounds under hydrothermal conditions. *Materials Research Society Symposium Proceedings*, 432:317–331, 1996. doi: 10.1557/PROC-432-317.

Boris Tagirov and Jacques Schott. Aluminum speciation in crustal fluids revisited. *Geochimica et Cosmochimica Acta*, 65(21):3965–3992, 2001. doi: 10.1016/S0016-7037(01)00705-0.