# Equilibrium in CHNOSZ

Jeffrey M. Dick

May 19, 2015

## 1 Background; terminology

*Species of interest* (or simply *species*) are chosen by the user to represent the possible species in a chemical system. *Basis species* combine linearly to make up the species of interest. Basis species are analogous to thermodynamic components in that they are the minimum number to describe the compositional variation, but unlike components basis species can be charged.

The calculations of chemical equilibrium in CHNOSZ are formulated for a system that is open with respect to the basis species. Therefore, the natural variables are temperature, pressure and chemical potentials of the basis species.

To calculate the equilibrium distribution of species in a given system, a single linear balancing constraint must be specified. Therefore, we say things like "the reactions are balanced on $CO_2$" or "the reactions are balanced on protein length". The *balancing coefficients* describe these constraints. At different times in the documentation of CHNOSZ, the balancing constraint has been associated with the conserved component, immobile component, or conserved basis species, all with the same meaning.

The "reactions" in the preceding statements refer to *transformations* between species. The actual calculations, however, start with the definitions of the formation reactions. The *formation reaction* for any species has one mole of the species as a product, and the mass balance is made up of the basis species.

By definition, the formation reaction is written to form one mole of a species. For many systems, the extent of the molar formula of the species is not a matter of concern. However, for systems made of polymers, such as proteins, it is often desirable to normalize the molar formula by the balancing coefficients. This normalization has been referred to previously as using the residue equivalents of proteins [2]; here the terminology of *normalize* will be used preferentially[1].

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 illustrated using specific example that has been described previously [2, 3] (the "CSG" example). The example system demonstrates that two approaches are equivalent when the molar formluas are normalized.

## 2 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 $\mu_i$ and $\mu_i^\circ$ stand for the chemical potential and the standard chemical potential of the $i$th species, and $R$ and $T$ represent the gas constant and the temperature in Kelvin. Chemical activity is related to molality ($m_i$) by

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

where $\gamma_i$ stands for the activity coefficient of the $i$th 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

[1] The older style of function call using `diagram(..., residue=TRUE)` has been replaced by `equilibrate(..., normalize=TRUE)` or `diagram(..., normalize=TRUE)` starting with version 0.9-9 of CHNOSZ.

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 proceding. The `add.obigt()` function does just that, as well as adds other species from the "supplemental" database provided with CHNOSZ:

```
> library(CHNOSZ)
> data(thermo)
> add.obigt()

add.obigt: using default file:
/tmp/RtmpzlIj8E/Rinst5a8f785e0fe0/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
```

# 3 Reaction-matrix approach

## 3.1 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  0     3590     -3    aq CSG_METVO
2 2555 1042 640  14 -2643.5  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 (which is the opposite of 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
1 CSG_METVO    553 C2575H4040.93490596228N645O884S11-56.0650940377185 -24880934 -56.06509
2 CSG_METJA    530 C2555H3976.12975396577N640O865S14-55.8702460342319 -24236262 -55.87025
        G.Z         ZC
1 -24976763 -0.1444660
2 -24413723 -0.1385519
```

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 [1]. 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] 107.6774

$`3591`
[1] 317.1877
```

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. $\boldsymbol{A}/2.303RT$.

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 $\ln 10$),

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

for Reactions 3 and 4 as

$$\begin{aligned} \boldsymbol{A}_3/2.303RT &= \log K_3 - \log Q_3 \\ &= \log K_3 + \log a_{stuff,3} - \log a_{\text{CSG\_METVO}} \\ &= \boldsymbol{A}^*_3/2.303RT - \log a_{\text{CSG\_METVO}} \end{aligned} \tag{6}$$

and

$$\begin{aligned} \boldsymbol{A}_4/2.303RT &= \log K_4 - \log Q_4 \\ &= \log K_4 + \log a_{stuff,4} - \log a_{\text{CSG\_METJA}} \\ &= \boldsymbol{A}^*_4/2.303RT - \log a_{\text{CSG\_METJA}}\,. \end{aligned} \tag{7}$$

The $\boldsymbol{A}^*$ 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 $\boldsymbol{A}^*_3/2.303RT = 104.6774$ and $\boldsymbol{A}^*_4/2.303RT = 314.1877$.

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 $i$th 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 $i$th protein ($a_i$) and the activity of the residue equivalent of the $i$th 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 $\boldsymbol{A} = \boldsymbol{A}_3/553 = \boldsymbol{A}_4/530$ is a condition for equilibrium. Combining this with Eqs. (6) and (7) gives

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

and

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

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: coefficients are protein length
equilibrate: balancing coefficients are 553 530
equilibrate: logarithm of total protein length is 0.0346284566253204

> e$loga.equil

[[1]]
[1] -225.9512

[[2]]
[1] -2.689647
```

Those are the logarithms of the equilibrium activities of the proteins. Combining these values with either Eqs. (10) or (11) gives us the same value for affinity of the formation reactions per residue (or per protein backbone group), $\boldsymbol{A}/2.303RT = 0.5978817$. 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.

## 3.2 Implementing the reaction-matrix approach

The implementation used in CHNOSZ for finding a solution to the system of equations 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 as $a_{\text{ic}}$ (e.g., the 1.083 in Eqn. 9).
2. Write a function for the logarithm of activity of each of the species of interest: $\boldsymbol{A} = \left(\boldsymbol{A}_i^* - 2.303RT \log a_i\right)/n_{\text{ic},i}$, where $n_{\text{ic},i}$ stands for the number of moles of the immobile component that react in the formation of one mole of the $i$th species. (e.g., for systems of proteins where the backbone group is conserved, $n_{\text{ic},i}$ is the same as $n_i$ in Eq. 8). Calculate values for each of the $\boldsymbol{A}_i^*$. Metastable equilibrium is implied by the identity of $\boldsymbol{A}$ in all of the equations.
3. Write a function for the total activity of the immobile component: $a'_{\text{ic}} = \sum n_{\text{ic},i} a_i$.
4. The difference function is now $\delta a_{\text{ic}} = a'_{\text{ic}} - a_{\text{ic}}$.

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

This approach is subject to failure if for all trial ranges of $A$ the $\delta a_{\mathrm{ic}}$ are of the same sign, which gives an error message like "i tried it 1000 times but can't make it work". Even if values of $\delta a_{\mathrm{ic}}$ 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 (currently hard-coded) tolerance is not reached.

### 3.3 CSG Example: normalized formulas (residue equivalents)

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

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

and

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

The formulas of the residue equivalents are those of the proteins divided by the number of residues in each protein. With the `protein.basis()` function it is possible to see 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,] 4.656420 1.934901 1.166365 0.01989150 -4.824593 -0.1013835
[2,] 4.820755 1.966038 1.207547 0.02641509 -4.987736 -0.1054156
```

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

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

and

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

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 $A^*_{12} = A_{12} + 2.303RT \log a_{\mathrm{CSG\_METVO(residue)}}$ and $A^*_{13} = A_{13} + 2.303RT \log a_{\mathrm{CSG\_METJA(residue)}}$. 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

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

and

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

which are equivalent to Equations 12 and 13 in the paper [2] but with more decimal places shown. A third equation arises from the conservation of amino acid residues:

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

The solution to these equations is $a_{\mathrm{CSG\_METVO(residue)}} = 0.3065982$, $a_{\mathrm{CSG\_METJA(residue)}} = 0.7764018$ 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 used by `diagram()` when `residue=TRUE` (which is the default setting). Instead, the Boltzmann distribution, described next, is implemented for that situation.

# 4 Boltzmann distribution

## 4.1 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.

# 5 Notes on implementation

## 5.1 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.

```
> # get an error if we don't data(thermo), only in the re-building vignettes of R CMD check
> data(thermo)
> protein <- iprotein(c("CSG_METVO", "CSG_METJA"))
> 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

> swap.basis("O2", "H2")

    C H N O S Z ispecies    logact state
CO2 1 0 0 2 0 0       69 -3.000000    aq
H2O 0 2 0 1 0 0        1  0.000000   liq
NH3 0 3 1 0 0 0       68 -4.000000    aq
H2S 0 2 0 0 1 0       70 -7.000000    aq
H2  0 2 0 0 0 0       66 -4.657486    aq
H+  0 1 0 0 0 1        3 -7.000000    aq

> protein.equil(protein, loga.protein=-3)

protein.equil: temperature from argument is 25 degrees C
protein.equil: pH from thermo$basis is 7
checkGHS: G of [Met] aq (1552) differs by 152 cal mol-1 from tabulated value
protein.equil: [Met] is from reference LD12
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 -47105102.0780865
```

```
protein.equil [1]: ionization reaction of protein has G0(cal/mol) of -95829.2021553493
protein.equil [1]: per residue, reaction to form ionized protein from basis species has G0/RT of -144.061868695781
protein.equil [1]: per residue, logQstar is 63.0052264992363
protein.equil [1]: per residue, Astar/RT = -G0/RT - 2.303logQstar is -1.01302662207411
check it!       per residue, Astar/RT calculated using affinity() is -1.01302662207413
protein.equil [all]: lengths of all proteins are 553 530
protein.equil [all]: Astar/RT of all residue equivalents are -1.01302662207411 -0.284203417393798
protein.equil [all]: sum of exp(Astar/RT) of all residue equivalents is 1.11573182734004
protein.equil [all]: equilibrium degrees of formation (alphas) of residue equivalents are 0.325453020153928 0.67454
check it!     alphas of residue equivalents from equilibrate() are 0.325453020153923 0.674546979846077
protein.equil [all]: for activity of proteins equal to 10^-3, total activity of residues is 10^0.0346284566253204
protein.equil [all]: log10 equilibrium activities of residue equivalents are -0.452883237292139 -0.136359341216436
protein.equil [all]: log10 equilibrium activities of proteins are -3.19560836859684 -2.86063521081723
check it!    log10 eq'm activities of proteins from equilibrate() are -3.19560836859684 -2.86063521081722
```

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.) If the checks failed, an error would be produced and this vignette could not be published on CRAN. Therefore, the calculations made using `affinity()` and `equilibrate()` are demonstrably consistent with the methods we have outline above.

## 5.2 Visualizing the effects of normalization

A comparison of the outcomes of equilibrium calculations that do and do not use the normalized formulas for proteins was given in a publication [2]. An expanded version of a diagram in that paper is below.

```
> organisms <- c("METSC", "METJA", "METFE", "HALJP",
+   "METVO", "METBU", "ACEKI", "GEOSE", "BACLI", "AERSA")
> proteins <- c(rep("CSG", 6), rep("SLAP", 4))
> 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(proteins, organisms)

    CO2  H2O  NH3 H2S      O2 H+ ispecies logact state       name
1  2812 1066  747  16 -2909.0  0     3392     -3    aq  CSG_METSC
2  2555 1042  640  14 -2643.5  0     3370     -3    aq  CSG_METJA
3  2815 1071  747  14 -2914.5  0     3393     -3    aq  CSG_METFE
4  3669 1367  971   0 -3608.5  0     3394     -3    aq  CSG_HALJP
5  2575 1070  645  11 -2668.0  0     3369     -3    aq  CSG_METVO
6  1362  519  355   4 -1400.5  0     3395     -3    aq  CSG_METBU
7  3584 1431  926   4 -3730.5  0     3396     -3    aq SLAP_ACEKI
8  5676 2320 1489   3 -5904.5  0     3397     -3    aq SLAP_GEOSE
9  3977 1594 1068   2 -4131.0  0     3398     -3    aq SLAP_BACLI
10 2250  861  618   2 -2322.5  0     3399     -3    aq SLAP_AERSA

> a <- affinity(O2=c(-100, -65))
```

```
energy.args: temperature is 25 C
energy.args: pressure is Psat
energy.args: variable 1 is log_f(O2) at 128 values from -100 to -65
subcrt: 16 species at 298.15 K and 1 bar (wet)
subcrt: 18 species at 298.15 K and 1 bar (wet)
```

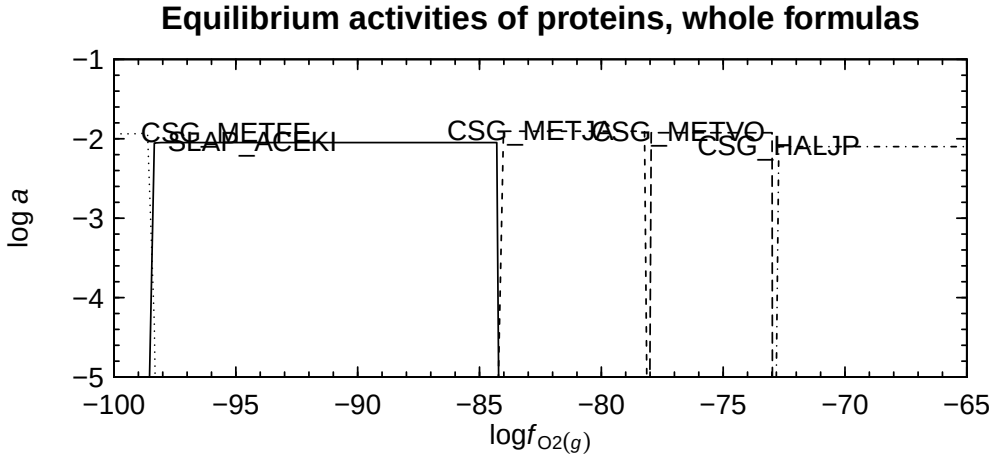
```
> par(mfrow=c(2, 1))
> e <- equilibrate(a)
```

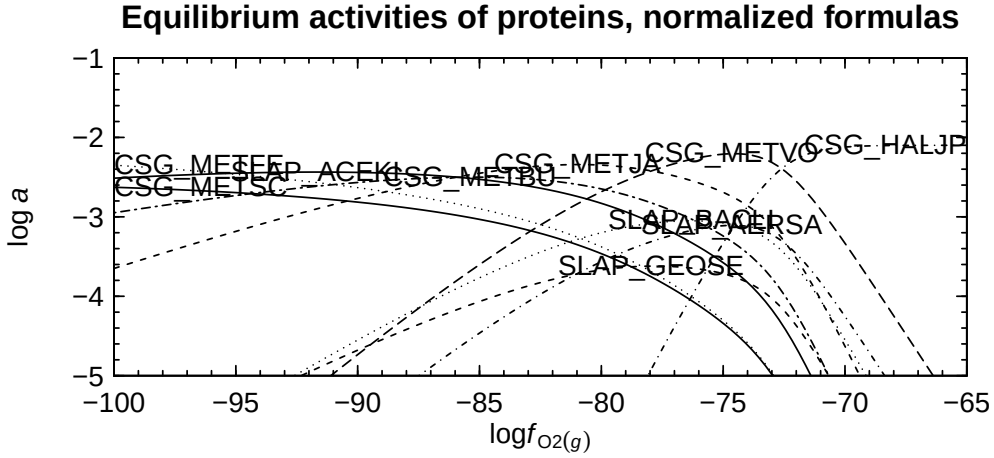
```
balance: coefficients are protein length
equilibrate: balancing coefficients are 571 530 571 828 553 278 736 1198 844 481
equilibrate: logarithm of total protein length is 0.81888541459401
```

```
> diagram(e, ylim=c(-5, -1), legend.x=NA)
> title(main="Equilibrium activities of proteins, whole formulas")
> e <- equilibrate(a, normalize=TRUE)
```

```
balance: coefficients are protein length
equilibrate: balancing coefficients are 571 530 571 828 553 278 736 1198 844 481
equilibrate: logarithm of total protein length is 0.81888541459401
equilibrate: using 'normalize' for molar formulas
```

```
> diagram(e, ylim=c(-5, -1), legend.x=NA)
> title(main="Equilibrium activities of proteins, normalized formulas")
```



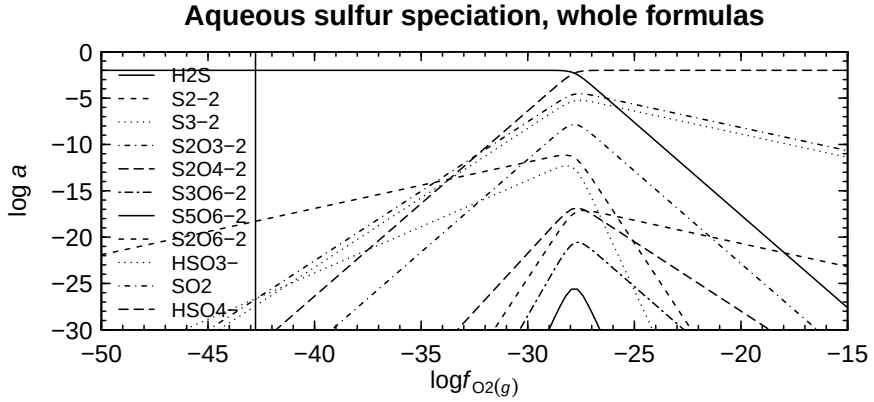
The reaction-matrix approach described above can also be applied to systems having conservation coefficients that differ from unity, such as many mineral and inorganic systems, where the immobile component has different molar coefficients in the formulas. For example, consider a system like that described in [4]:

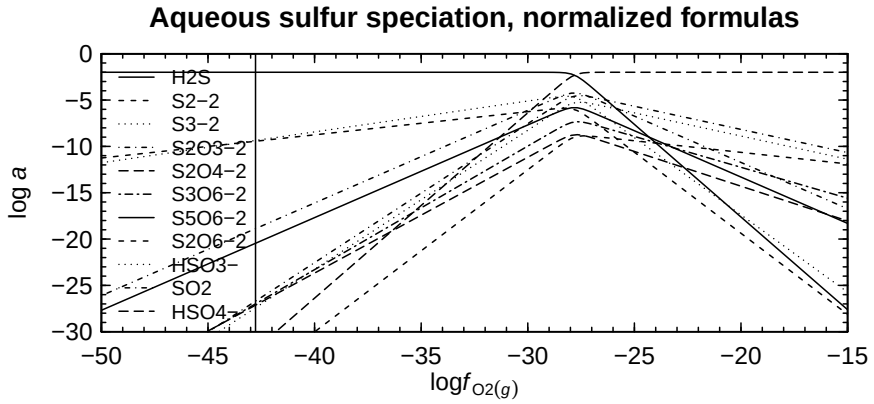
«»=

```
> 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

> basis("pH",5)

    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     -5    aq

> species(c("H2S", "S2-2", "S3-2", "S2O3-2", "S2O4-2",
+   "S3O6-2", "S5O6-2", "S2O6-2", "HSO3-", "SO2", "HSO4-"))

   CO2 H2O NH3 H2S  O2 H+ ispecies logact state   name
1    0   0   0   1 0.0  0       70     -3    aq    H2S
2    0  -1   0   2 0.5 -2       53     -3    aq   S2-2
3    0  -2   0   3 1.0 -2       54     -3    aq   S3-2
4    0  -1   0   2 2.0 -2       26     -3    aq S2O3-2
5    0  -1   0   2 2.5 -2     1072     -3    aq S2O4-2
6    0  -2   0   3 4.0 -2     1077     -3    aq S3O6-2
7    0  -4   0   5 5.0 -2     1079     -3    aq S5O6-2
8    0  -1   0   2 3.5 -2     1076     -3    aq S2O6-2
9    0   0   0   1 1.5 -1       23     -3    aq  HSO3-
10   0  -1   0   1 1.5  0       78     -3    aq    SO2
11   0   0   0   1 2.0 -1       25     -3    aq  HSO4-

> a <- affinity(O2=c(-50, -15), T=325, P=350)

energy.args: temperature is 325 C
energy.args: pressure is 350 bar
energy.args: variable 1 is log_f(O2) at 128 values from -50 to -15
subcrt: 17 species at 598.15 K and 350 bar (wet)

> par(mfrow=c(2, 1))
> e <- equilibrate(a, loga.balance=-2)

balance: coefficients are moles of H2S in formation reactions
equilibrate: balancing coefficients are 1 2 3 2 2 3 5 2 1 1 1
equilibrate: logarithm of total moles of H2S (from loga.balance) is -2

> diagram(e, ylim=c(-30, 0), legend.x="topleft", cex.names=0.8)
> title(main="Aqueous sulfur speciation, whole formulas")
> e <- equilibrate(a, loga.balance=-2, normalize=TRUE)
```

```
balance: coefficients are moles of H2S in formation reactions
equilibrate: balancing coefficients are 1 2 3 2 2 3 5 2 1 1 1
equilibrate: logarithm of total moles of H2S (from loga.balance) is -2
equilibrate: using 'normalize' for molar formulas
```

```
> diagram(e, ylim=c(-30, 0), legend.x="topleft", cex.names=0.8)
> title(main="Aqueous sulfur speciation, normalized formulas")
```




The first diagram is quantitatively very similar to the one shown by Seewald, 1997, but if we use the normalized formulas, in this case divided by $H_2S$ in the formation reactions, the range of activities of species is lower at any given $\log f_{O_{2(g)}}$. Maybe `normalize=TRUE` doesn't make sense for systems like this where the formulas of species are similar in size to those of the basis species. For biomacromolecules such as proteins it seems to be a useful concept.

With the potential for calculating equilibrium activities of proteins comes the desire to compare these calculations to actual measurements! To be continued...

# 6 The maximum affinity method

When making a predominance diagram, we don't need to know the equilibrium activities of all species in the system, only the species that has the higest activity at any temperature, pressure and chemical activities of basis species. The *maximum affinity method* for producing predominance diagrams existed in early versions of CHNOSZ, before equilibrium calculations were implemented.

In a system where whole formulas are used in the formation reactions, derivation of the maximum affinity method is easy. Consider a generalized reaction to form a species of interest $Z_i$ from basis species $X$ and $Y$:

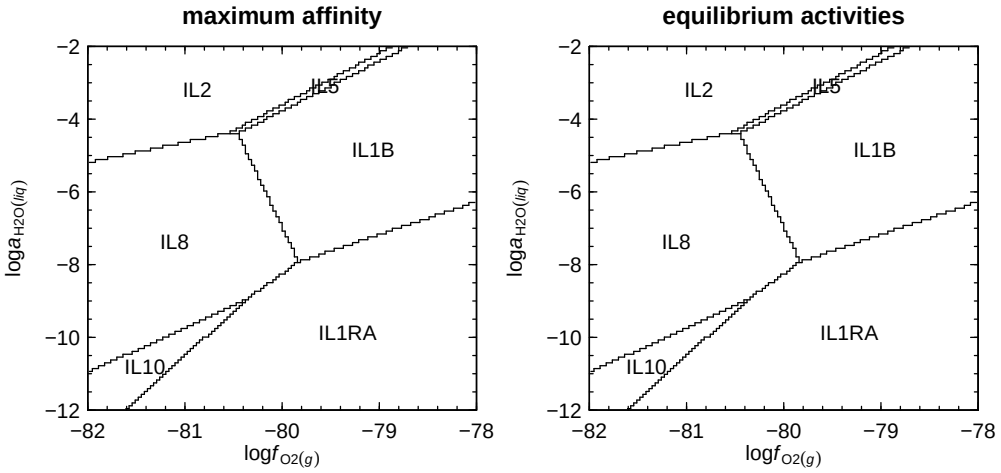
$$n_{X,i}X + n_{Y,i}Y \rightleftharpoons Z_i\,, \tag{20}$$

where $n_{X,i}$ and $n_{Y,i}$ stand for the reaction coefficients on the basis species.

...

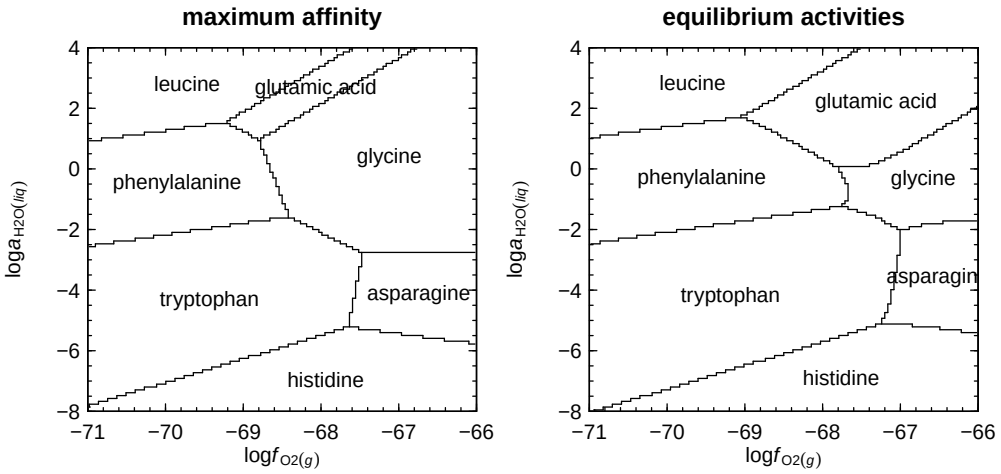
So for `normalize=TRUE` the maximum affinity method results in an identical predominance diagram to one calculated using the equilibrium activities (Blood plasma proteins, "IL" for interleukin):

```
> data(thermo)  # cleanup from previous plot
> basis(c("CO2", "NH3", "H2S", "H2O", "oxygen"), c(-3, -3, -10))
> f <- system.file("extdata/abundance/AA03.csv", package="CHNOSZ")
> pdat <- read.csv(f, as.is=TRUE)
> iil <- grep("^IL", pdat$name)
> species(pdat$name[iil], "HUMAN")
> a <- affinity(O2=c(-82, -78), H2O=c(-12, -2))
> par(mfrow=c(1, 2))
> dA <- diagram(a, normalize=TRUE, main="maximum affinity")
> e <- equilibrate(a, normalize=TRUE)
> dE <- diagram(e, main="equilibrium activities")
> stopifnot(identical(dA$predominant, dE$predominant))
```




Here is an example where the predominant species in the equilibrium assemblage are ***not*** identical to those calculated the maximum affinity method, and it is not possible for the maximum affinity method to make those curved lines!!

```
> basis("CHNOS+")
> species(aminoacids(""))
> a <- affinity(O2=c(-71, -66), H2O=c(-8, 4))
> par(mfrow=c(1, 2))
> dA <- diagram(a, main="maximum affinity")
> e <- equilibrate(a)
> dE <- diagram(e, main="equilibrium activities")
```




Take-home: when making predominance diagrams, confidently use the maximum affinity method when `normalize=TRUE` (as done here for proteins); otherwise it is advisable to compute the equilibrium distribution.

# 7 Document revision history

- 2009-11-29 Initial version containing CSG example (title: Calculating relative abundances of proteins)
- 2012-09-30 Renamed from previous "protactiv.Rnw": Remove activity comparisons, add maximum affinity method.

# References

[1] J. M. Dick, D. E. LaRowe, and H. 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.

[2] 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.

[3] 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.

[4] J. 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.