Hot-spring proteins in CHNOSZ

Jeffrey M. Dick and Everett L. Shock

March 16, 2020

1 Introduction

This document is intended to demonstrate the calculations described in two recent papers [1, 2] dealing with the distribution and abundances of proteins in "Bison Pool", a hot spring in Yellowstone National Park. The calculations use metastable equilibrium to interrelate the compositions of proteins (from metagenomic data) with environmental conditions, particularly pH, temperature, and redox chemistry. This document is focused on the details of the calculations and has little in the way of introduction of concepts or interpretation and discussion; please see the papers for more details.

There is no hidden code in this document; all code used to make the figures is shown in the blocks, but much of the text output (particularly in the figure-generating code) has been suppressed. The R code here is based on the Supporting Information for the papers, with modifications. In order to keep the code here as short and efficient as possible, some aspects of the published figures including manual labeling and higher resolution are not all reproduced; these changes are noted where possible. Also, some shortcuts (e.g. using `xtabs()` to create the BLAST frequency table, and `revisit()` to calculate the ΔG_{tr}) have been introduced here in order to streamline the code.

Load CHNOSZ and the thermodynamic database. In order to reproduce the calculations from the 2011 paper, we load old values of standard Gibbs energy and enthalpy of the methionine sidechain group from [3]; these are inaccurate values that were updated by [4] and are available starting in CHNOSZ_0.9-9 (the current values are used further below). This step also loads values for the glycine group and the protein backbone [UPBB] that were superseded in 2019.

```
library(CHNOSZ)
reset()

## reset:  resetting "thermo" object
## obigt:  loading default database with 1849 aqueous, 3374 total species

add.obigt("OldAA")

## add.obigt:  read 64 rows; made 21 replacements, 43 additions [energy units:  cal]
## add.obigt:  use obigt() or reset() to restore default database
```

Some values that are shared among different calculations: measured temperature (°C) and pH. These are representative values only; the actual values are not constant but vary due to water flow, weather, animals, etc.

```
bison.T <- c(93.3, 79.4, 67.5, 65.3, 57.1)
bison.pH <- c(7.350, 7.678, 7.933, 7.995, 8.257)
```

Now let's plot the measured temperature and pH in the hot spring. Distances (in meters) of the sampling sites are measured from the source of the hot spring. The `Tfun()` and `pHfun()` are also used further below.

```r
distance <- c(0, 6, 11, 14, 22)
par(mfrow=c(1, 2), mar=c(4, 4, 3, 2))
xpoints <- seq(0, 22, length.out=128)
# T plot
plot(distance, bison.T, xlab="distance, m", ylab=axis.label("T"))
Tfun <- splinefun(distance, bison.T, method="mono")
lines(xpoints, Tfun(xpoints))

# pH plot
plot(distance, bison.pH, xlab="distance, m", ylab="pH")
pHfun <- splinefun(distance, bison.pH, method="mono")
lines(xpoints, pHfun(xpoints))
```
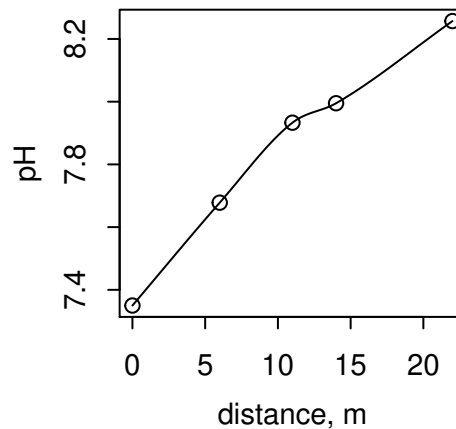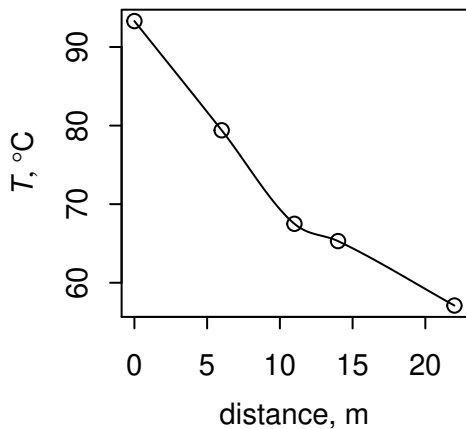

[DOI link](#) to published figure (panels A and B). The length of `xpoints` here, 128, is the resolution used for the figures in the 2011 paper (defined in the `mkargs()` function of the Supporting Information).

2 General setup

Load the proteins. These are "model proteins", i.e. average amino acid compositions of sequences classified according to their functional annotation or by their phylum association, at each of the five sampling sites located in the hot spring.

```r
# read the amino acid compositions
aa.annot <- read.csv(system.file("extdata/protein/DS11.csv", package="CHNOSZ"), as.is=TRUE)
aa.phyla <- read.csv(system.file("extdata/protein/DS13.csv", package="CHNOSZ"), as.is=TRUE)
```

Here are the site names for the sampling locations (also referred to as sites 1–5).

```r
sites <- c("N", "S", "R", "Q", "P")
sitenames <- paste("bison", sites, sep="")
```

Here are the classifications according to functional annotation:

```
classes <- unique(aa.annot$protein)
classes

##  [1] "overall"       "transferase"   "transport"     "dehydrogenase"
##  [5] "synthase"      "ATPase"        "kinase"        "synthetase"
##  [9] "membrane"      "permease"      "hydrolase"     "oxidoreductase"
## [13] "transcription" "peptidase"     "ribosomal"     "protease"
## [17] " reductase"    "transposase"   "periplasmic"   "signal"
## [21] "phosphatase"
```

Here are names of phyla and colors and line types used for plotting (colors based on Figure 2 of Wu and Eisen, 2008 [5], with modifications):

```
# the names of the phyla in alphabetical order (except Deinococcus-Thermus at end)
phyla.abc <- sort(unique(aa.phyla$organism))[c(1:7,9:11,8)]
# an abbreviation for Dein.-Thermus
phyla.abbrv <- phyla.abc
phyla.abbrv[[11]] <- "Dein.-Thermus"
phyla.cols <- c("#f48ba5", "#f2692f", "#cfdd2a",
  "#962272", "#87c540", "#66c3a2", "#12a64a", "#f58656",
  "#ee3237", "#25b7d5", "#3953a4")
phyla.lty <- c(1:6, 1:5)
phyla.abbrv

##  [1] "Acidobacteria"  "Aquificae"      "Bacteroidetes"  "Chlorobi"
##  [5] "Chloroflexi"    "Crenarchaeota"  "Cyanobacteria"  "Euryarchaeota"
##  [9] "Firmicutes"     "Proteobacteria" "Dein.-Thermus"
```

3 Average oxidation state of carbon

The average oxidation state of carbon ($\overline{Z}_C$) is calculated using

$$\overline{Z}_C = \frac{Z - n_H + 2\,(n_O + n_S) + 3n_N}{n_C}\,, \tag{1}$$

where n_C, n_H, n_N, n_O and n_S are the numbers of the indicated subscripted elements in the chemical formula of a protein or other chemical species, and Z is the net charge of the chemical species.

```
# 2011 plot
ylab <- expression(bar(italic(Z))[C])
plot(0, 0, xlim=c(-0.5, 5), ylim=c(-0.27, -0.11), xlab="location", xaxt="n", ylab=ylab)
axis(1, at=1:5)
col <- c("green", rep("black", 20))
lwd <- c(3, rep(1, 20))
clab <- c("hydrolase", "overall", "protease", "oxidoreductase", "transport", "membrane", "permease")
pf.annot <- protein.formula(aa.annot)
ZC.annot <- ZC(pf.annot)
for(i in 1:length(classes)) {
  lines(1:5, ZC.annot[(1:5)+5*(i-1)], col=col[i], lwd=lwd[i])
  if(classes[i] %in% clab) text(0.8, ZC.annot[1+5*(i-1)], classes[i], adj=1)
}

title(main="annotations")
```

```
# 2013 plot
pf.phyla <- protein.formula(aa.phyla)
ZC.phyla <- ZC(pf.phyla)
# set up plot
plot(0, 0, xlim=c(1, 5), ylim=c(-0.27, -0.11), xlab="location", ylab=ylab)
for(i in 1:length(phyla.abc)) {
  # which of the model proteins correspond to this phylum
  iphy <- which(aa.phyla$organism==phyla.abc[i])
  # the locations (of 1, 2, 3, 4, 5) where this phylum is found
  ilocs <- match(aa.phyla$protein[iphy], sitenames)
  # the plotting symbol: determined by alphabetical position of the phylum
  points(ilocs, ZC.phyla[iphy], pch=i-1, cex=1.2)
  # a line to connect same phyla occurring at adjacent sites
  inlocs <- rep(NA, 5)
  inlocs[ilocs] <- ilocs
  lines(inlocs, ZC.phyla[iphy][match(1:5, ilocs)])
}
legend("bottomright", pch=0:10, legend=phyla.abbrv, bg="white", cex=0.9)
title(main="major phyla")
```
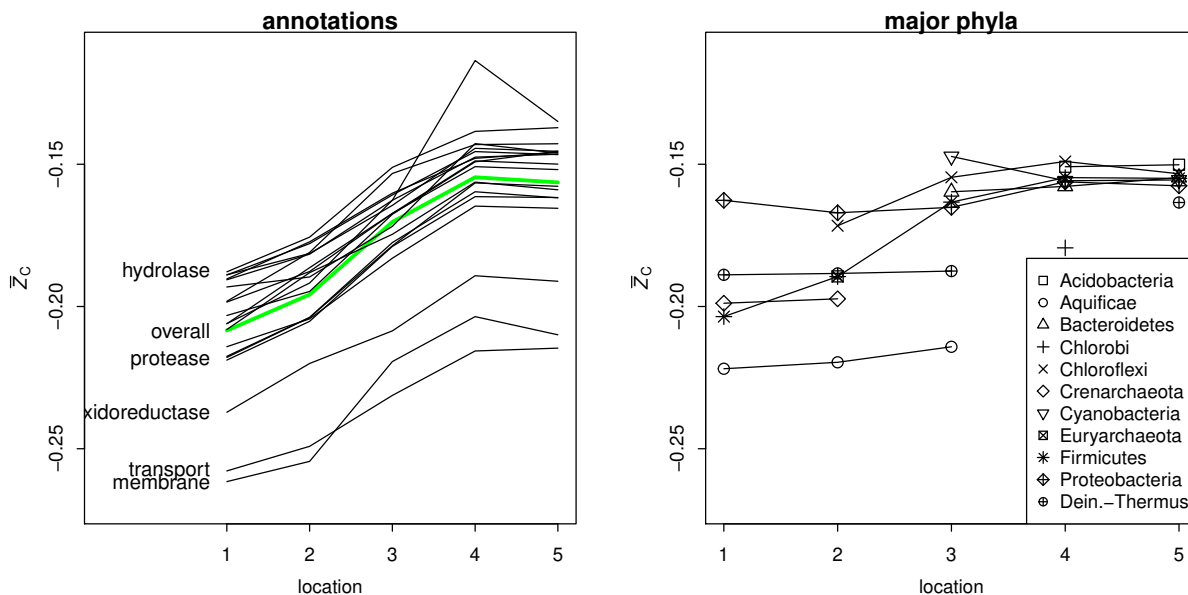

DOI links for these plots as they appeared in the publications: plot 1, plot 2. Compared to the papers, a different y-axis scale is used here, in order to have same scale on both plots; permease at lower $\overline{Z}_C$ does not appear in plot 1 here.

4 Relative stabilities of "overall" model proteins

4.1 Formation reactions of proteins from basis species

Function to setup the basis species. The basis species consist of HCO_3^-, H_2O, NH_3, HS^-, H_2 and H^+ (all aqueous species except for H_2O liquid).

```
setup.basis <- function() {
  basis(c("HCO3-", "H2O", "NH3", "HS-", "H2", "H+"))
  basis(c("HCO3-", "NH3", "HS-", "H+"), c(-3, -4, -7, -7.933))
```

```
}
```

Set up the basis species and the species. Here, we add the proteins using the previously read amino acid compositions (`aa.annot`), and we save their index number (`ip.annot`) for use later. For now, we load the species corresponding to the "overall" model proteins (the first 5 in aa.annot).

```
setup.basis()

##        C H N O S  Z ispecies logact state
## HCO3- 1 1 0 3 0 -1      13 -3.000   aq
## H2O   0 2 0 1 0  0       1  0.000  liq
## NH3   0 3 1 0 0  0      64 -4.000   aq
## HS-   0 1 0 0 1 -1      22 -7.000   aq
## H2    0 2 0 0 0  0      62  0.000   aq
## H+    0 1 0 0 0  1       3 -7.933   aq

ip.annot <- add.protein(aa.annot)
```

add.protein: added 105 new protein(s) to thermo$protein

```
species("overall", sitenames)

##       HCO3-     H2O    NH3    HS-      H2      H+ ispecies logact state            name
## 1 1020.55 -2772.8 270.30 5.694 2147.5 1026.24     3418     -3    aq overall_bisonN
## 2  935.79 -2541.8 251.89 5.381 1963.2  941.17     3419     -3    aq overall_bisonS
## 3  981.83 -2666.2 271.40 5.362 2047.3  987.20     3420     -3    aq overall_bisonR
## 4  952.44 -2583.0 266.32 5.834 1978.5  958.27     3421     -3    aq overall_bisonQ
## 5  941.70 -2554.4 263.07 5.750 1957.0  947.45     3422     -3    aq overall_bisonP
```

The first 6 columns there represent the stoichiometry of the reactions to form the proteins from the basis species. The reactions can be divided by the lengths (number of amino acid residues) of the proteins to write per-residue formation reactions.

```
pl <- protein.length(ip.annot[1:5])
mysp <- species()
mysp[, 1:6]/pl

##      HCO3-     H2O    NH3      HS-     H2     H+
## 1 5.1250 -13.924 1.3574 0.028594 10.784 5.1536
## 2 5.0763 -13.788 1.3664 0.029190 10.650 5.1054
## 3 5.0226 -13.639 1.3884 0.027429 10.473 5.0500
## 4 4.9657 -13.467 1.3885 0.030417 10.315 4.9961
## 5 4.9720 -13.487 1.3890 0.030359 10.333 5.0023
```

Note e.g. the higher coefficient on H_2 for sites 1 and 2; increasing the activity of H_2 (more reducing conditions) has a relatively more favorable mass-action effect on the formation of the proteins at the higher temperature sites.

4.2   Chemical affinities along a chemical gradient

Function to calculate $\log a_{H_2}$ as a linear equation in T. This was used to fit the spatial distribution of proteins in the 2011 paper [1] (also shown as Equation 2 in 2013 [2]).

$$\log a_{H_{2(aq)}} = -11 + 3/40 \times T(°C) \tag{2}$$

```
get.logaH2 <- function(T) -11 + T * 3/40
```

Calculate the residue-normalized chemical affinities of the formation reactions of the overall model proteins. First set activities of the proteins equal to unity (logarithm of activity equal to zero). Then calculate affinities per mole of protein for the temperature, pH and $\log a_{H_2(aq)}$ of each site. Use the lengths of the model proteins to calculate the affinities per residue.

```
species(1:5, 0)
```

```
##       HCO3-      H2O    NH3   HS-      H2      H+ ispecies logact state          name
## 1 1020.55 -2772.8 270.30 5.694 2147.5 1026.24     3418      0    aq overall_bisonN
## 2  935.79 -2541.8 251.89 5.381 1963.2  941.17     3419      0    aq overall_bisonS
## 3  981.83 -2666.2 271.40 5.362 2047.3  987.20     3420      0    aq overall_bisonR
## 4  952.44 -2583.0 266.32 5.834 1978.5  958.27     3421      0    aq overall_bisonQ
## 5  941.70 -2554.4 263.07 5.750 1957.0  947.45     3422      0    aq overall_bisonP
```

```
a <- affinity(T=bison.T, pH=bison.pH, H2=get.logaH2(bison.T))
```

```
## affinity:   pressure is Psat
## affinity:   variable 1 is T at 5 values from 330.25 to 366.45 K
## affinity:   variable 2 is pH at 5 values from 7.35 to 8.257
## affinity:   variable 3 is log10(a_H2) at 5 values from -6.7175 to -4.0025
## subcrt:  11 species at 5 values of T (C) and P (bar) (wet) [energy units:  cal]
## subcrt:  18 species at 5 values of T (C) and P (bar) (wet) [energy units:  cal]
```

```
a.res <- t(as.data.frame(a$values))/pl
a.res
```

```
##           [,1]    [,2]    [,3]    [,4]    [,5]
## X3418 -18.713 -27.891 -35.386 -36.823 -42.266
## X3419 -18.838 -27.911 -35.318 -36.739 -42.120
## X3420 -19.113 -28.051 -35.349 -36.749 -42.052
## X3421 -19.262 -28.078 -35.276 -36.657 -41.889
## X3422 -19.246 -28.075 -35.284 -36.667 -41.907
```

DOI link to published table.

The affinities are expressed as dimensionless values, i.e. $A/2.303RT$ where A, R and T stand for chemical affinity, gas constant, and temperature in Kelvin. The affinities are all negative, but show a progression from higher for protein (row) 1 at the conditions of sites (columns) 1 and 2 to higher for protein (row) 4 at the conditions of sites (columns) 3 to 5.

```
apply(a.res, 2, which.max)
```

```
## [1] 1 1 4 4 4
```

To go from residue-normalized affinities to the actual progression of relative stabilities of the proteins (i.e. size-adjusted affinities), we subtract the logarithm of the protein length from the per-residue affinities (also see description following Eq. 19 in Ref. [1]).

```
a.res <- a.res - log10(pl)
apply(a.res, 2, which.max)
```

```
## [1] 1 2 4 4 4
```

This 1-2-4 progression of stabilities is visualized below; for some groups of proteins, an update to the thermodynamic properties of the methionine sidechain causes proteins from site 3 also to become stable.

4.3 Relative stabilities along a chemical gradient

Set the temperature limits (`Tlim`) over which to perform the calculations.

```
Tlim <- c(50, 100)
```

   Make two plots here. (1) The metastable equilibrium predominance diagram of the overall model proteins as a function of temperature and logarithm of activity of hydrogen. The stability fields for the proteins from the higher temperatures are at higher activities of hydrogen in the diagram. The dotted line passes through the stability fields of the model proteins at approximately the actual environmental temperatures. (2) Combine the gradients of temperature, pH and hydrogen activity to calculate the metastable equilibrium activities of the proteins. The total activity of residues is set by reference activities of the proteins equal to 10^{-3}. In order to label the x-axis "distance", modify a couple of entries in the list returned by `affinity()` (`vars`, `vals`); otherwise the x-axis would represent temperature (the first variable in the argument list to `affinity()`).

```
par(mfrow=c(1, 2))
# first plot
a <- affinity(T=Tlim, H2=c(-7, -4))
diagram(a, fill=NULL, names=as.character(1:5), normalize=TRUE)
lines(Tlim, get.logaH2(Tlim), lty=3)
# second plot
species(1:5, -3)
xT <- Tfun(xpoints)
xpH <- pHfun(xpoints)
xH2 <- get.logaH2(xT)
a <- affinity(T=xT, pH=xpH, H2=xH2)
a$vars[1] <- "distance, m"
a$vals[[1]] <- xpoints
e <- equilibrate(a, normalize=TRUE)
diagram(e, legend.x=NULL)
legend("bottom", lty=1:5, legend=1:5, bty="n", cex=0.6)
```
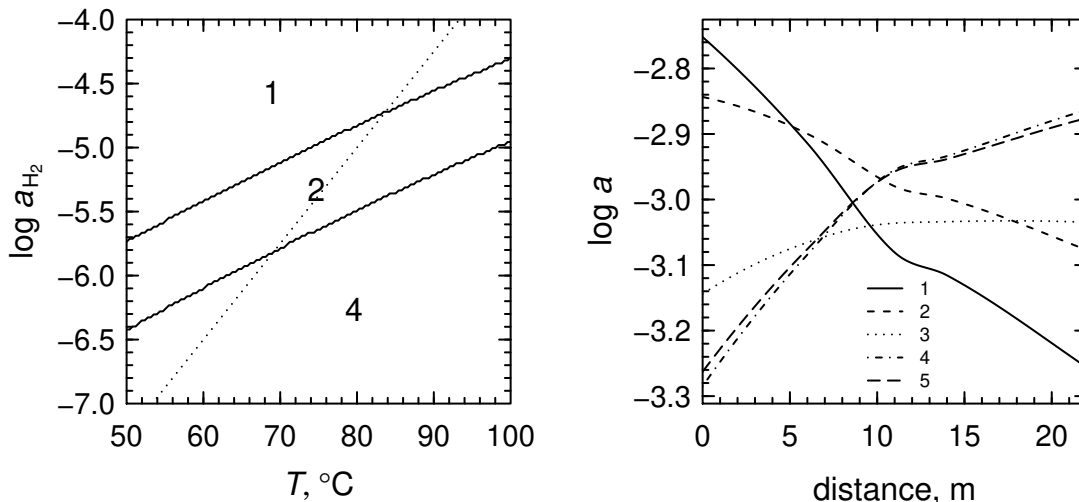

[DOI link](#) to published figure (panels B and F).

Plot the equilibrium degrees of formation as a function of distance for different classes of proteins. Calculate the affinities for all of the proteins. For each group of three, make strip charts using the `strip()` function in CHNOSZ. The heights of the bars represent the relative abundances of the model proteins. The five steps of the color code go from site 1 (red) to site 5 (blue).

```
loadclass <- function(class) {
  species(delete=TRUE)
  species(rep(class, each=5), rep(sitenames, length(class)))
}
xclasses <- c("overall", "transferase", "transport", "synthetase", "membrane", "permease")
loadclass(xclasses)
a <- affinity(T=xT, pH=xpH, H2=xH2)
a$vars[1] <- "distance, m"
a$vals[[1]] <- xpoints
col <- c("red", "orange", "yellow", "green", "blue")
par(mfrow=c(1, 2), mar=c(4, 4, 1, 1))
for(i in 1:2) {
  ispecies <- lapply((1:3)+(i-1)*3, function(x) {1:5+(x-1)*5} )
  names(ispecies) <- xclasses[(1:3)+(i-1)*3]
  strip(a = a, ispecies = ispecies, col = col, xticks = distance, cex.names = 1)
}
```
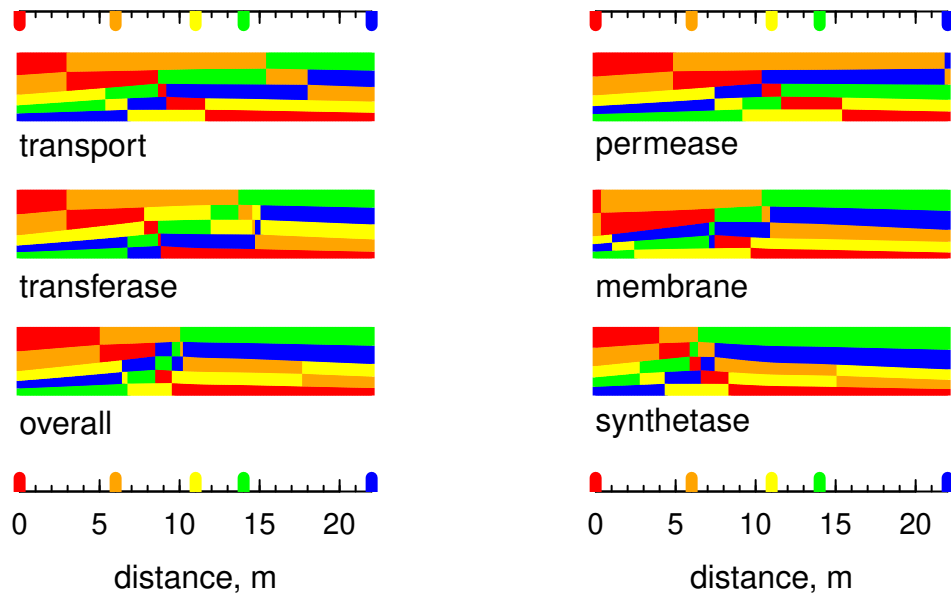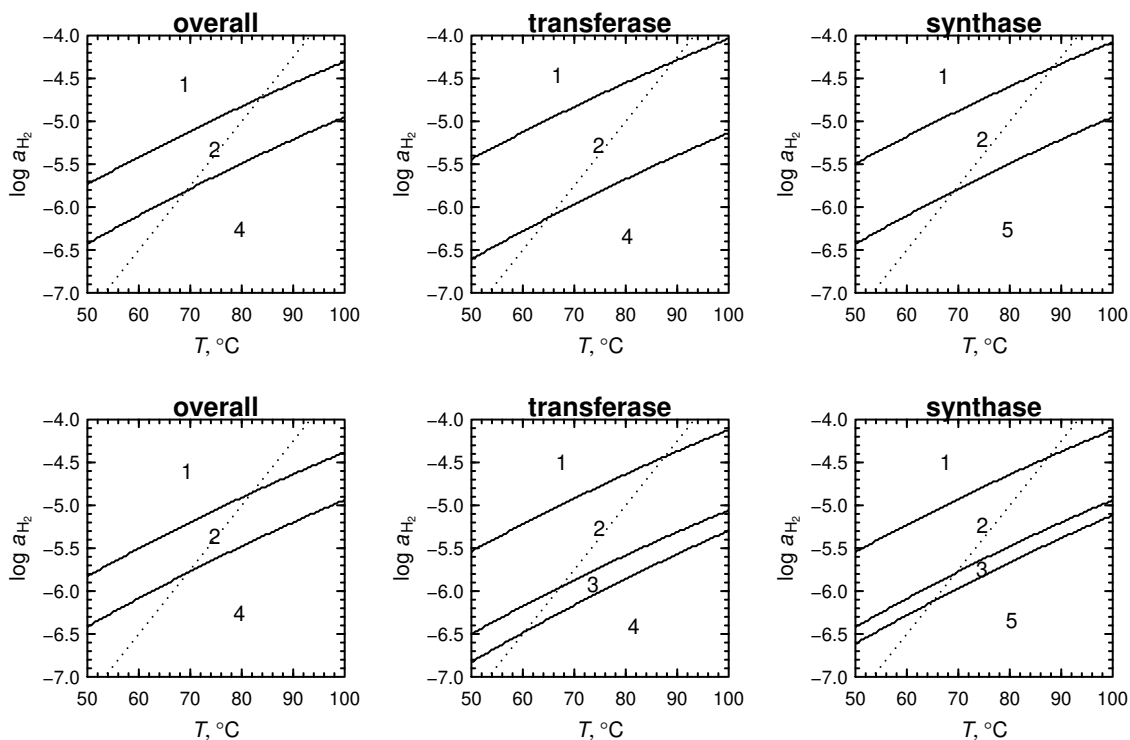

DOI link to published figure.

5 Comparing old and new methionine sidechain parameters

Make some $T - \log a_{H_2}$ metastable equilibrium predominance diagrams using different values for the thermodynamic properties of the methionine sidechain group [Met]. The first row shows the results using the old values (defined on page 1); for the second row (j=2), we reset the database to use the current (revised) parameters.

```
par(mfrow=c(2, 3))
for(j in 1:2) {
  # use old [Met] for first row and new [Met] for second row
  if(j==2) {
    reset()
    add.obigt("OldAA", c("[Gly]", "[UPBB]"))
    ip.annot <- add.protein(aa.annot)
  }
  # setup basis species and proteins
  setup.basis()
  # make the plots
  for(annot in c("overall", "transferase", "synthase")) {
    ip <- ip.annot[aa.annot$protein==annot]
    a <- affinity(T=c(50, 100), H2=c(-7, -4), iprotein=ip)
    diagram(a, fill=NULL, names=as.character(1:5), normalize=TRUE)
    # add logaH2-T line
    lines(par("usr")[1:2], get.logaH2(par("usr")[1:2]), lty=3)
    # add a title
    title(main=annot)
  }
}
```


DOI link to published figure, which was made with higher resolution (256) and manual placement of some labels.

9

6 Relative abundance calculations

Function to return the fractional abundances based on BLAST counts, stored in the `ref` column of `aa.phyla`.

```r
alpha.blast <- function() {
  out <- xtabs(ref ~ protein + organism, aa.phyla)
  # put it in correct order, then turn counts into fractions
  out <- out[c(1,5:2), c(1:7,9:11,8)]
  out <- out/rowSums(out)
  return(out)
}
```

Function to calculate metastable equilibrium degrees of formation of proteins (normalized to residues) as a function of $\log a_{H_2}$ for a specified location. Note: in the call to `affinity()`, increase the resolution from 101 to 1001 to reproduce the calculations in paper.

```r
alpha.equil <- function(i=1) {
  # order the names and counts to go with the alphabetical phylum list
  iloc <- which(aa.phyla$protein==sitenames[i])
  iloc <- iloc[order(match(aa.phyla$organism[iloc], phyla.abc))]
  # set up basis species, with pH specific for this location
  setup.basis()
  basis("pH", bison.pH[i])
  # calculate metastable equilibrium activities of the residues
  a <- affinity(H2=c(-11, -1, 101), T=bison.T[i], iprotein=ip.phyla[iloc])
  e <- equilibrate(a, loga.balance=0, as.residue=TRUE)
  # remove the logarithms to get relative abundances
  a.residue <- 10^sapply(e$loga.equil, c)
  colnames(a.residue) <- aa.phyla$organism[iloc]
  # the BLAST profile
  a.blast <- alpha.blast()
  # calculate Gibbs energy of transformation (DGtr) and find optimal logaH2
  iblast <- match(colnames(a.residue), colnames(a.blast))
  r <- revisit(e, "DGtr", log10(a.blast[i, iblast]), plot.it=FALSE)
  # return the calculated activities, logaH2 range, DGtr values, and optimal logaH2
  return(list(alpha=a.residue, H2vals=a$vals[[1]], DGtr=r$H, logaH2.opt=r$xopt))
}
```

Now we're ready to make a plot! We start by adding the proteins with amino acid composition that were read from the file. Then plot the degrees of formation as a function of $\log a_{H_2}$ at sites 1, 3 and 5, but record the results (including calculated relative abundances and optimal values of $\log a_{H_2}$) at all 5 sites.

```r
ip.phyla <- add.protein(aa.phyla)
layout(matrix(1:6, ncol=3), heights=c(2, 1))
equil.results <- list()
for(i in 1:5) {
  # get the equilibrium degrees of formation and the optimal logaH2
  ae <- alpha.equil(i)
  equil.results[[i]] <- ae
  if(i %in% c(1, 3, 5)) {
    iphy <- match(colnames(ae$alpha), phyla.abc)
    # top row: equilibrium degrees of formation
    thermo.plot.new(xlim=range(ae$H2vals), ylim=c(0, 0.5), xlab=axis.label("H2"),
      ylab=expression(alpha[equil]), yline=2, cex.axis=1, mgp=c(1.8, 0.3, 0))
```

```
    for(j in 1:ncol(ae$alpha)) {
      lines(ae$H2vals, ae$alpha[, j], lty=phyla.lty[iphy[j]])
      ix <- seq(1, length(ae$H2vals), length.out=11)
      ix <- head(tail(ix, -1), -1)
      points(ae$H2vals[ix], ae$alpha[, j][ix], pch=iphy[j]-1)
    }
    title(main=paste("site", i))
    legend("topleft", pch=iphy-1, lty=phyla.lty[iphy], legend=phyla.abbrv[iphy], bg="white")
    # bottom row: Gibbs energy of transformation and position of minimum
    thermo.plot.new(xlim=range(ae$H2vals), ylim=c(0, 1/log(10)), xlab=axis.label("H2"),
      ylab=expr.property("DGtr/2.303RT"), yline=2, cex.axis=1, mgp=c(1.8, 0.3, 0))
    lines(ae$H2vals, ae$DGtr)
    abline(v=ae$logaH2.opt, lty=2)
    abline(v=get.logaH2(bison.T[i]), lty=3, lwd=1.5)
    if(i==1) legend("bottomleft", lty=c(3, 2), lwd=c(1.5, 1), bg="white",
      legend=c("Equation 2", "optimal"))
  }
}
```
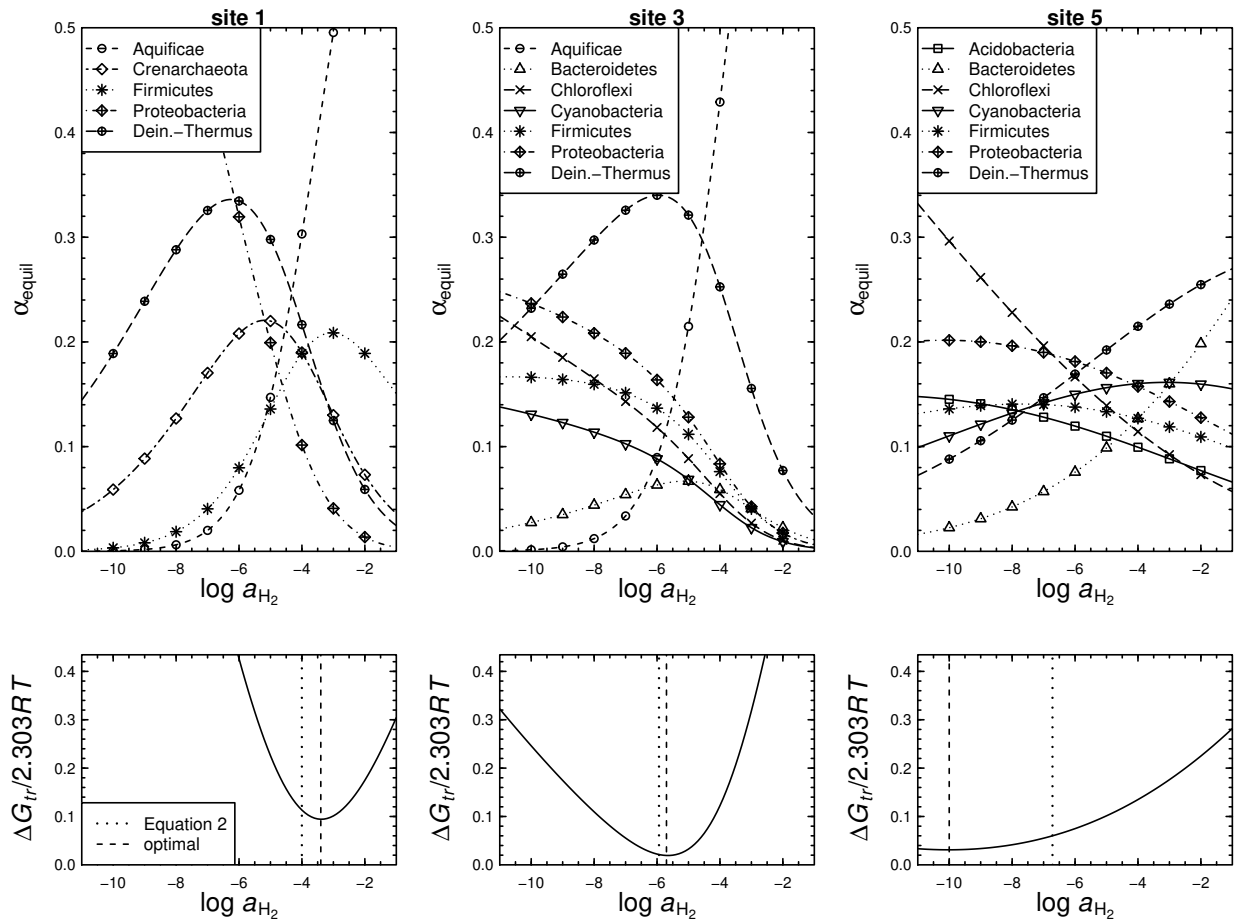

DOI link to published version of the figure.

7 Activity of hydrogen comparison

Let's compare the computed activities of hydrogen with various redox indicators measured in the field.

7.1 Conversion of field measurements

ORP Oxidation-reduction potential (ORP), measured in the field using a portable probe and pH/voltmeter, can be converted to Eh by adding the potential of the reference electrode, in this case silver-silver chloride (Ag/AgCl) in saturated KCl. As an approximation, the following equation from Ref. [6] for Ag/AgCl (1M KCl) is used, with temperature in °C and potential in volts.

```
E.AgAgCl <- function(T) {
  0.23737 - 5.3783e-4 * T - 2.3728e-6 * T^2 - 2.2671e-9 * (T+273)
}
```

Data from Bison Pool and another hot spring shown in Fig. 9 of Ref. [1] are both included, but for simplicity here they are lumped together. The ORP values have units of mV.

```
T.ORP <- c(93.9, 87.7, 75.7, 70.1, 66.4, 66.2)
pH.ORP <- c(8.28, 8.31, 7.82, 7.96, 8.76, 8.06)
ORP <- c(-258, -227, -55, -58, -98, -41)
```

Convert ORP to effective values of $\log a_{H_{2(aq)}}$. First convert to Eh (in volts). Then convert Eh to pe. Then combine Eh and pH with the law of mass action for

$$2e^- + 2H^+ \rightleftharpoons H_{2(aq)}$$
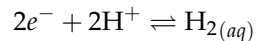
to calculate $\log a_{H_{2(aq)}}$. The "law of mass action" is the equality between the equilibrium constant (K) and the activity product (Q) of the species in the reaction.

```
Eh <- ORP/1000 + E.AgAgCl(T.ORP)
pe <- convert(Eh, "pe", T=convert(T.ORP, "K"))
logK.ORP <- subcrt(c("e-", "H+", "H2"), c(-2, -2, 1), T=T.ORP)$out$logK
logaH2.ORP <- logK.ORP - 2*pe - 2*pH.ORP
```

Sulfide/Sulfate For sulfide/sulfate, assign activities that are equal to concentrations (in molal units) measured in the field season of 2005, and use the law of mass action for

$$HS^- + 4H_2O \rightleftharpoons SO_4{}^{-2} + H^+ + 4H_{2(aq)}$$
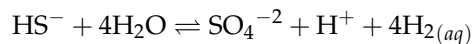
to calculate an effective activity of hydrogen. Sulfide was determined spectrophotometrically at the hot spring, and sulfate was determined using ion chromatography on water samples returned from the field.

```
loga.HS <- log10(c(4.77e-6, 2.03e-6, 3.12e-7, 4.68e-7, 2.18e-7))
loga.SO4 <- log10(c(2.10e-4, 2.03e-4, 1.98e-4, 2.01e-4, 1.89e-4))
logK.S <- subcrt(c("HS-", "H2O", "SO4-2", "H+", "H2"), c(-1, -4, 1, 1, 4), T=bison.T)$out$logK
logaH2.S <- (logK.S + bison.pH - loga.SO4 + loga.HS) / 4
```

Dissolved Oxygen Calculate the effective activity of hydrogen corresponding to the dissolved oxygen concentration using the law of mass action for

$$0.5O_{2(aq)} + H_{2(aq)} \rightleftharpoons H_2O\,.$$
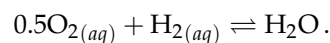
Convert the dissolved oxygen concentrations (mg/L) to molarities (mol/L) to set the activity of $O_{2(aq)}$.

```
DO <- c(0.173, 0.776, 0.9, 1.6, 2.8)
logaO2 <- log10(DO/1000/32)
logK <- subcrt(c("O2", "H2", "H2O"), c(-0.5, -1, 1), T=bison.T)$out$logK
logaH2.O <- 0 - 0.5*logaO2 - logK
```

Plot it! The points in the left plot represent field-based indicators for redox potential, while the points in the right plot are the optimal values of $\log a_{H_2}$ for the microbial phylum abundance model. The various redox indicators are not in equilibrium with each other (the $\log a_{H_2}$ values are different), but they all show an increase in apparent $\log a_{H_2}$ with temperature. The dotted line in both plots portrays Equation (2).

```
# 2011 plot
xlab <- axis.label("T")
ylab <- axis.label("H2")
plot(Tlim, get.logaH2(Tlim), xlim=Tlim, ylim=c(-45,0),
  xlab=xlab, ylab=ylab, type="l", lty=3)
points(T.ORP, logaH2.ORP, pch=15)
lines(T.ORP, logaH2.ORP, lty=2)
points(bison.T, logaH2.O, pch=16)
lines(bison.T, logaH2.O, lty=2)
points(bison.T, logaH2.S, pch=17)
lines(bison.T, logaH2.S, lty=2)
llab <- c("Equation 2", "ORP", "dissolved oxygen", "sulfate/sulfide")
text(c(65, 80, 80, 74), c(-4, -25, -40, -11), llab)

# 2013 plot
plot(Tlim, get.logaH2(Tlim), xlim=Tlim, ylim=c(-11,-2),
  xlab=xlab, ylab=ylab, type="l", lty=3)
lines(bison.T, sapply(equil.results, "[", "logaH2.opt"), lty=2)
points(bison.T, sapply(equil.results, "[", "logaH2.opt"), pch=21, bg="white")
text(90, -5.3, "Equation 2")
text(66, -9, "optimal parameterization\nfor observed\nphylum abundances", adj=0)
```
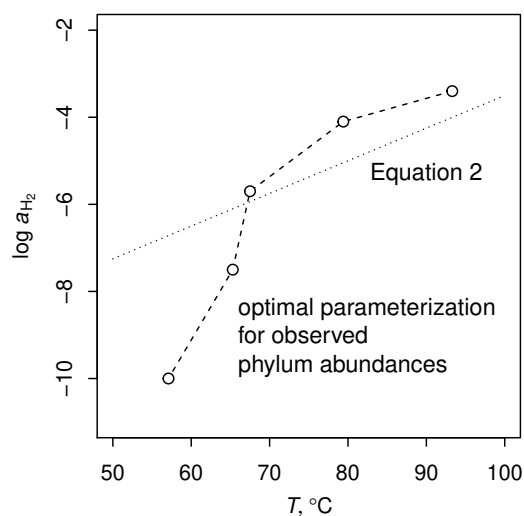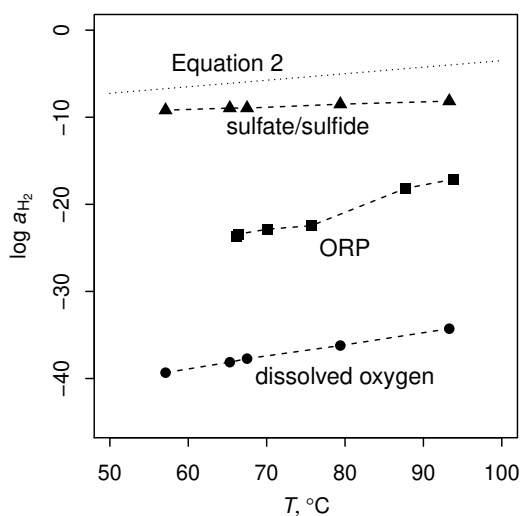

DOI links to published versions: plot 1, plot 2.

8 Relative abundance comparison

Let's make some plots comparing the relative abundances calculated above (stored in `equil.results`) with the abundances from BLAST counts. We'll also store the results of the optimized models in `equil.opt`.

```
layout(matrix(c(1, 2, 3, 4, 5, 6), nrow=2, byrow=TRUE), widths=c(2, 2, 2))
par(mar=c(2.5, 0, 2.5, 0))
plot.new()
legend("topright", pch=0:11, legend=phyla.abbrv, bty="n", cex=1.5)
lim <- c(-6, -0.5)
equil.opt <- a.blast <- alpha.blast()
for(iloc in 1:5) {
  a.equil <- equil.results[[iloc]]
  iopt <- match(a.equil$logaH2.opt, a.equil$H2vals)
  ae.opt <- a.equil$alpha[iopt, ]
  # which are these phyla in the alphabetical list of phyla
  iphy <- match(names(ae.opt), phyla.abc)
  equil.opt[iloc, iphy] <- ae.opt
  mar <- c(2.5, 4.0, 2.5, 1)
  thermo.plot.new(xlab=expression(log[2]*alpha[obs]), ylab=expression(log[2]*alpha[equil]),
    xlim=lim, ylim=lim, mar=mar, cex=1, yline=1.5)
  # add points and 1:1 line
  points(log2(a.blast[iloc, iphy]), log2(ae.opt), pch=iphy-1)
  lines(lim, lim, lty=2)
  title(main=paste("site", iloc))
  # within-plot legend: DGtr
  DGexpr <- as.expression(quote(Delta*italic(G[tr])/italic(RT) == phantom()))
  DGval <- format(round(2.303*a.equil$DGtr[iopt], 3), nsmall=3)
  legend("bottomright", bty="n", legend=c(DGexpr, DGval))
}
```
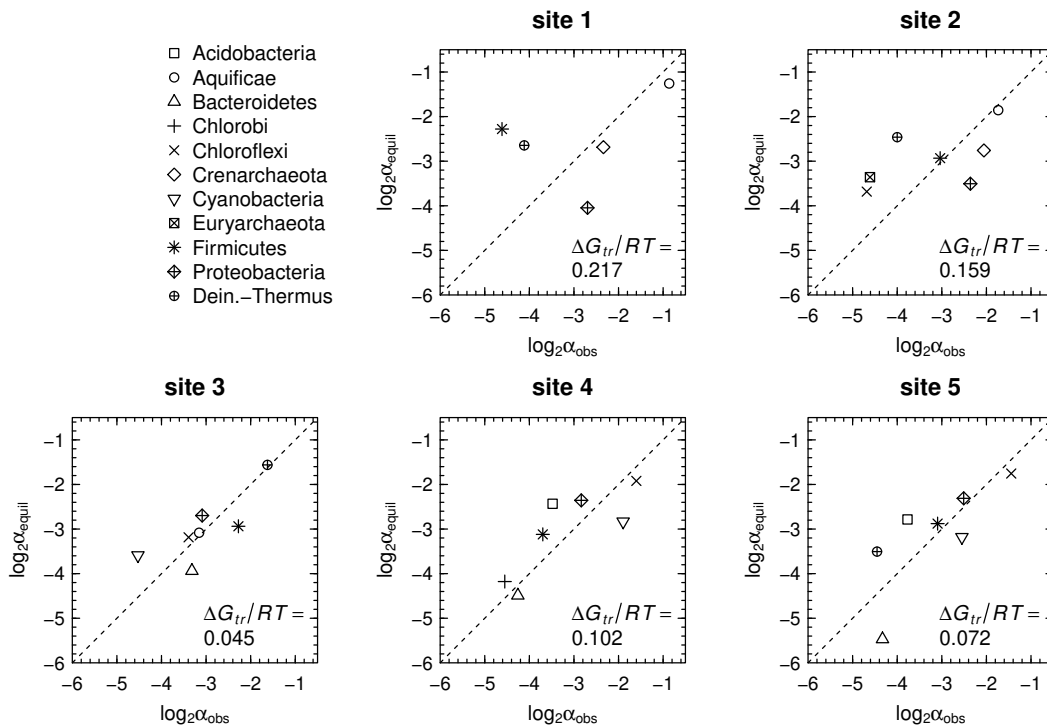
Finally, a comparison in the form of two barplots.

```r
par(mar=c(4, 4, 3, 0), mgp=c(1.8, 0.7, 0))
par(mfrow=c(1, 3), cex=1)
# make the blast plot
ab <- alpha.blast()
rownames(ab) <- 1:5
barplot(t(ab), col=phyla.cols, ylab=NULL, xlab="site", axes=TRUE, cex.axis=0.8, cex.names=0.8, las=1)
mtext(expression(alpha[obs]), 2, 2, cex=1.1*par("cex"))
title(main="BLAST profile", cex.main=0.8)
# make the equilibrium plot
rownames(equil.opt) <- 1:5
barplot(t(equil.opt), col=phyla.cols, ylab=NULL, xlab="site", axes=TRUE, cex.axis=0.8,
  cex.names=0.8, las=1)
mtext(expression(alpha[equil]), 2, 2, cex=1.1*par("cex"))
title(main="metastable\nequilibrium", cex.main=0.8)
# add legend
par(mar=c(4, 1, 3, 0))
plot.new()
legend("bottomleft", legend=rev(phyla.abbrv), fill=rev(phyla.cols), bty="n", cex=0.7)
```
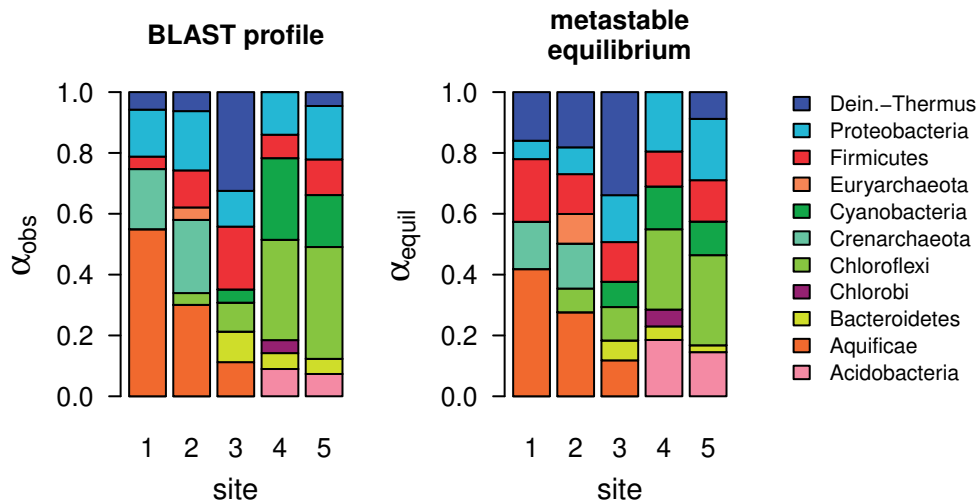

[DOI link](#) to published figure.

9 Document history

- 2011-08-23 Initial version (CHNOSZ_0.9-7)

- 2012-01-07 Use `add.obigt()` for superseded properties of methionine sidechain group (changed to `mod.obigt()` on 2016-12-21).

- 2014-02-01 Include results from 2013 paper and change VignetteEngine to knitr.

References

[1] 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.

[2] 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.

[3] 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.

[4] Douglas E. LaRowe and Jeffrey M. Dick. Calculation of the standard molal thermodynamic properties of crystalline peptides. *Geochimica et Cosmochimica Acta*, 80:70–91, 2012. doi: 10.1016/j.gca.2011.11.041.

[5] Martin Wu and Jonathan A. Eisen. A simple, fast, and accurate method of phylogenomic inference. *Genome Biology*, 9(10):R151, 2008. ISSN 1474-760X. doi: 10.1186/gb-2008-9-10-r151.

[6] A. J. Bard, R. Parsons, and J. Jordan. *Standard Potentials in Aqueous Solution*. M. Dekker, New York, 1985. URL http://www.worldcat.org/oclc/12106344.