Supplementary Protocol S1 for

Dependency Map of Proteins in the Small Ribosomal Subunit

Kay Hamacher[1,2], Joanna Trylska[1,2,3], J. Andrew McCammon[1,2,4]

[1] Department of Chemistry and Biochemistry, University of California at San Diego, 9500 Gilman Drive, La Jolla, CA 92093-0365, USA

[2] Center for Theoretical Biological Physics, University of California at San Diego, 9500 Gilman Drive, La Jolla, CA 92093-0374, USA

[3] Interdisciplinary Centre for Mathematical and Computational Modelling, Warsaw University, Pawińskiego 5a, 02-106 Warsaw, Poland

[4] Howard Hughes Medical Institute and Department of Pharmacology, University of California at San Diego, La Jolla, CA 92093-0365, USA

Correspondence should be addressed to:
Kay Hamacher,
Center for Theoretical Biological Physics and
Department of Chemistry and Biochemistry, MC 0374
University of California at San Diego,
9500 Gilman Drive, La Jolla, CA 92093-0374, USA
e-mail: hamacher_at_ctbp.ucsd.edu
phone +1 (858) 534 - 7256
fax +1 (858) 534 - 7697

1 General Definitions

$S = \{S2, S3, \ldots, S20, THX\}$ is the set of all ribosomal proteins. G_M is the free energy as obtained from a SCPCP-calculation for a system containing the rRNA and the proteins of set M.

2 One Protein Removal

The algorithm in a pseudo-programming language reads

For each $i \in S$
  Compute $\Delta G^i_{\text{bind}} = G_S - G_{S\setminus\{i\}} - G_{\{i\}}$
Construct a vector of ranks $\vec{r}$ according to the respective ΔG^i_{bind}

For each $i \in S$
  $S^{(i)} = S\setminus\{i\}$
  For each $j \in S^{(i)}$
    Compute $\Delta G^{(i),j}_{\text{bind}} = G_{S^{(i)}} - G_{S^{(i)}\setminus\{j\}} - G_{\{j\}}$
  Construct a vector of ranks $\vec{r}^{(i)}$ according to the respective $\Delta G^{(i),j}_{\text{bind}}$

Construct the sets $I^{(i)}$ of all j that were pushed backwards in
 the ranking by $\Delta_r^{(i),j}$ places in comparison to $\vec{r}$.
 These sets are the (by i) influenced proteins.

The obtained $\Delta_r^{(i),j}$ are the weights on the arrows in figure 2 of the main manuscript. The arrows point from a protein i that (negatively) influences the ranking of a protein j.

3 Two Protein Removal

The algorithm in a pseudo-programming language reads

For each $i \in S$
  $S^{(i)} = S\setminus\{i\}$
  For each $j \in S^{(i)}$
    $S^{(i,j)} = S\setminus\{i,j\}$
    For each $k \in S^{(i,j)}$
      Compute the tensor element $\Delta G^k_{i,j} = G_{S^{(i,j)}} - G_{S^{(i,j)}\setminus\{k\}} - G_{\{k\}}$
      Set $\Delta G^k_{ii} = \Delta G^i_{ij} = \Delta G^i_{ji} = 0$

For each $k \in S$
  Diagonalize the (20×20)-matrix ΔG^k and obtain
   the eigenvalues λ_l and eigenvectors $\vec{u}_l$.
  Obtain for $l \in \{1, 20\}$ the most probable entry-value U_l in $\vec{u}_l$.
  Construct the sets I^k_1 and I^k_{20} whose elements are given by the a
   index of the $\vec{u}_l$ for which $\left|\frac{\vec{u}_{1,a}}{U_1}\right| > 1.01$ and $\left|\frac{\vec{u}_{20,a}}{U_{20}}\right| > 1.10$, respectively.

The sets I_1^k and I_{20}^k contain the proteins that influence k. In figure 4 of the main text we again draw arrows from a protein a to the influenced protein k. The set I_{20}^k contains also non-local effects as discussed in the main text.

For the diagonalization procedure we applied the SVD-algorithm in the version described by Press et al. This is necessary - as mentioned in the main text - because of the additional null mode. For a general application we note that the procedure becomes more accurate for larger systems (larger number of chains) as small deviations in ΔG^k (e.g. due to numerics or parametrization) tend to be less relevant the larger the matrices ΔG^k.

Note that the matrices ΔG_{ij}^k are symmetric and one can therefore save a factor of 2 in CPU-time by reducing the loop over j to $j > i$ with $Si > Sj \iff i > j$ and setting $G_{ij} = G_{ji}$ afterwards.