Addtional material for Massingham and Goldman:
All Your Base: a fast and accurate probabilistic approach to base calling

Rapid calculation of $\widehat{M}$ and $\widehat{P}$ for the AYB base-caller

The statistical model fitted by AYB to the observed intensities from the Illumina platform is described by

$$I_i = \lambda_i M S_i P + N + \epsilon_i \tag{1}$$

where I_i is the observed intensities for cluster i, λ_i the cluster luminesence, M is the cross-talk, S_i the underlying sequence of the cluster, P the phasing, and N the systematic noise; see the text of the main paper for details. The least squares estimates for M and P are

$$\widehat{M}^t = \left(\sum_i w_i B_i B_i^t - \frac{1}{\widetilde{w}}\widetilde{B}\,\widetilde{B}^t\right)^{-1}\left(\sum_i w_i B_i I_i^t - \frac{1}{\widetilde{w}}\widetilde{B}\,\widetilde{I}^t\right) \tag{2}$$

$$\widehat{P} = \left(\sum_i w_i W_i^t W_i - \frac{1}{\widetilde{w}}\widetilde{W}^t\,\widetilde{W}\right)^{-1}\left(\sum_i w_i W_i^t I_i - \frac{1}{\widetilde{w}}\widetilde{W}^t\,\widetilde{I}\right) \tag{3}$$

where

$$\widetilde{w} = \sum_i w_i, \quad \widetilde{B} = \sum_i w_i B_i, \quad \widetilde{W} = \sum_i w_i W_i \text{ and } \widetilde{I} = \sum_i w_i I_i$$

but there is not a closed form solution for both simultaneously and it must be found using iterative techniques. Both these equations involve the summation, over all clusters, of terms consisting of several matrix multiplications and so calculation is extremely slow for densely covered tiles. By pre-calculating certain values, it is possible to remove the summations from both equations so that the iteration to find the least squares solution for M and P is quicker. In addition, it is possible to modify the iteration so that each step solves the least squares estimate of N as well as either M or P. For brevity we only derive the case for M and N here; the derivation of the simultaneous solution for P and N is similar.

The statistical model for the emitted intensities, equation 1 , can be rewritten as:

$$I_i = M' S_i' + \epsilon_i$$

by defining

$$M' = (M \quad N) \quad \text{and} \quad S_i' = \begin{pmatrix} \lambda_i S_i P \\ \mathrm{I_D} \end{pmatrix}$$

Tim Massingham, European Bioinformatics Institute where $\mathrm{I_D}$ is the identity matrix of appropriate size. The weighted least squares solution for M', the value $\widehat{M'}$ that minimises $\sum_i w_i \operatorname{tr}((I_i - M'S'_i)^t(I_i - M'S'_i))$, where $\operatorname{tr}(A)$ is the trace of matrix A, is

$$\widehat{M'}^t = \left(\sum_i w_i S'_i {S'}^t_i\right)^{-1} \left(\sum_i w_i S'_i I^t_i\right)$$

and the two summations involved can be expanded as

$$\sum_i w_i S'_i {S'}^t_i = \begin{pmatrix} \sum_i w_i \lambda_i^2 S_i P P^t S_i^t & \widetilde{S} P \\ P^t \widetilde{S}^t & \widetilde{w}\, \mathrm{I_D} \end{pmatrix} \qquad (4)$$

$$\sum_i w_i S'_i I^t_i = \begin{pmatrix} \sum_i w_i \lambda_i S_i P I_i^t \\ \widetilde{I}^t \end{pmatrix} \qquad (5)$$

where $\widetilde{S} = \sum_i w_i \lambda_i S_i$, $\widetilde{w} = \sum_i w_i$ and $\widetilde{I} = \sum_i w_i I_i$, all three of which are independent of M, P and N and so constant throughout the iteration. The two remaining terms, the top left block of the matrix in equation 4 and the top block of equation 5, still depend on P and so will vary between steps of the iteration.

The following identity[1] for matrices A, B and C:

$$\operatorname{vec}(ABC) = \left(C^t \otimes A\right) \operatorname{vec}(B)$$

where the operator 'vec' stacks the columns of a matrix on top of each other to form a vector and '$\otimes$' is the Kronecker matrix product. This identity can be applied to the two remaining summations in equations 4 and 5 to factor out the dependence on P and so express them as the product of a function of P and a summation independent of P:

$$\begin{aligned} \operatorname{vec}\left(\sum_i w_i \lambda_i^2 S_i P P^t S_i^t\right) &= \left[\sum_i w_i \lambda_i^2 \left(S_i \otimes S_i\right)\right] \operatorname{vec}\left(PP^t\right) \\ &= J \operatorname{vec}\left(PP^t\right) \quad \text{by appropriate definition of } J \\ \operatorname{vec}\left(\sum_i w_i \lambda_i S_i P I_i^t\right) &= \left[\sum_i w_i \lambda_i \left(I_i \otimes S_i\right)\right] \operatorname{vec}\left(P\right) \\ &= K \operatorname{vec}\left(P\right) \quad \text{by appropriate definition of } K \end{aligned}$$

Defining the '$\mathrm{Reshape}$' operator to be the inverse of vec, folding a vector back into a matrix of appropriate dimension, equations 4 and 5 can be expressed in terms P and values which are constant throughout the iteration and can therefore be pre-calculated:

$$\begin{aligned}
\sum_i w_i S'{}_i S'^t_i &= \begin{pmatrix} \mathrm{Reshape}\left(J\,\mathrm{vec}(PP^t)\right) & \tilde{S}P \\ P^t\tilde{S}^t & \tilde{w}\,\mathrm{I_D} \end{pmatrix} \\
\sum_i w_i S'_i I^t_i &= \begin{pmatrix} \mathrm{Reshape}\left(K\,\mathrm{vec}(P)\right) \\ \tilde{I}^t \end{pmatrix}
\end{aligned}$$

References

[1] Minka TP: **Old and New Matrix Algebra Useful for Statistics** 2000, [[http://research.microsoft.com/en-us/um/people/minka/papers/matrix/]]. [(accessed: 20 Oct. 2010)].