File S2 Solving the recursions numerically

Since all the recursions have the same general form, a generic method for solving them numerically will now be given. The idea is to use standard linear algebra methods to solve the standard linear system $A\underline{t} = \underline{b}$. Let $\underline{t} = (t_0, t_1, \ldots, t_{n+m})$ denote the vector of quantities we are solving for, where we order them according to number of active lines. For any given number n of active blocks and m of inactive blocks, so the current total number of blocks is $n + m$, write $\underline{t} = (t_0, t_1, \ldots, t_{n+m})$ where $t_i \equiv t_{i,n+m-i}$, and write $\ell := n + m$. Let $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ denote square $(\ell+1) \times (\ell+1)$ matrices whose rows and columns are enumerated from 0; with non-zero terms $a_{i-1,i} = \alpha_{i,\ell-i}$, $b_{i,i-1} = \beta_{i,\ell-i}$, $c_{i,i+1} = \gamma_{i,\ell-i}$, and let $\boldsymbol{I}$ denote a $(\ell+1) \times (\ell+1)$ identity matrix. Assume, by way of example, we are solving the recursion (10) for expected time to most recent common ancestor. Define the vector $\underline{k}$ with elements $k_i = 1/\lambda_{i,\ell-i}$, and $\underline{r} = (0, r_0, r_1, \ldots, r_{n+m-1})$ where $r_j = t_{j,n+m-j-1}$.

The recursion in Proposition (S1.1) can now be written

$$\underline{t} = \underline{k} + \boldsymbol{A}\underline{r} + (\boldsymbol{B} + \boldsymbol{C})\underline{t}$$

Assuming we solve for $\underline{t}$ iteratively, starting from $n+m = 2$, $\underline{r}$ is a vector of known constants; hence

$$\underline{s} = (\boldsymbol{I} - \boldsymbol{B} - \boldsymbol{C})^{-1}(\underline{k} + \boldsymbol{A}\underline{r})$$

where $\boldsymbol{I} - \boldsymbol{B} - \boldsymbol{C}$ should be non-singular and $(\boldsymbol{I} - \boldsymbol{B} - \boldsymbol{C})^{-1}$ easily computable.

Similar methods may be applied to the other recursions.