Inertia of HSS matrices using STRUMPACK
Jonas Actor\textsuperscript{a}, Pieter Ghysels\textsuperscript{b}, and Xiaoye Li\textsuperscript{b}

\textsuperscript{a}Computational and Applied Mathematics, Rice University
\textsuperscript{b}Scalable Solvers Group, Lawrence Berkeley National Laboratory

\textbf{Compression}

STRUMPACK compresses a \textit{symmetric indefinite} matrix $A$ (e.g. a KKT matrix) into HSS form.

$$A = \begin{bmatrix} D_1 & U_1^T B_{21} U_2^T \\ U_2^T B_{21} U_2 & D_2 \end{bmatrix} = \begin{bmatrix} D_1 & U_5^T B_{21} U_1^T \\ U_5 U_2^T B_{21} U_1^T & D_5 \end{bmatrix} \begin{bmatrix} U_3 & U_4^T B_{21} U_2^T \\ U_4 U_5^T B_{21} U_3^T & D_6 \end{bmatrix}$$

The \textit{inertia} $\nu(A)$, the count of positive, negative, and zero eigenvalues, is found via bottom-up traversal of the HSS tree.

\textbf{Introduce Zeros}

Use $\Omega$, matrices from STRUMPACK’s HSS factorization to annihilate subblocks, with $\Omega U_r = \begin{bmatrix} 0 \\ I \end{bmatrix}$.

By applying $\text{diag}(\Omega_r : \tau \text{ level } \ell)$ to the left and right of $A$, inertia is preserved.

\textbf{Partial LDL$^T$ Factorization}

Split into four subblocks $\Omega, D, \Omega^T = \begin{bmatrix} D_{r,11} & D_{r,12} \\ D_{r,21} & D_{r,22} \end{bmatrix}$. Take LDL$^T$ factorization of the top left subblock:

$$L_{r,11} \tilde{D}_r L_{r,11}^T = \text{LDL}(D_{r,11})$$

\textbf{Schur Complement}

Form the Schur complement using the LDL$^T$ factors. Use the Schur complement to form an LDL$^T$ factorization of all four subblocks:

$$L_{r,21} = D_{r,21} (\tilde{D}_r L_{r,11}^T)^{-1}$$

$$S_r = D_{r,22} - D_{r,21} D_{r,11}^{-1} D_{r,12}$$

$$\begin{bmatrix} D_{r,11} & D_{r,12} \\ D_{r,21} & D_{r,22} \end{bmatrix} = \begin{bmatrix} L_{r,11} & I \\ L_{r,21} & I \end{bmatrix} \begin{bmatrix} \tilde{D}_r & \hat{S}_r \\ \hat{S}_r^T & \nu(S) \end{bmatrix} \begin{bmatrix} L_{r,11} & I \\ L_{r,21} & I \end{bmatrix}^T$$

Note inertia is preserved: $\nu(D_r) = \nu(\Omega, D, \Omega^T) = \nu(\tilde{D}_r) + \nu(S_r)$.

\textbf{Repeat}

Consolidate the remaining factors by merging a factorized node $\sigma_2$ with its sibling node $\sigma_1$. Assign the consolidated factors to their parent node $\tau$ and repeat, thereby moving up the HSS tree.

$$D_r \leftarrow \begin{bmatrix} S_{\sigma_1} & B_{\sigma_1 \sigma_2} \\ B_{\sigma_2 \sigma_1} & S_{\sigma_2} \end{bmatrix}.$$