Links are disabled because this is a static copy of a profile report

total_hamiltonian (Calls: 25, Time: 2.686 s)

Generated 05-Aug-2024 09:51:40 using performance time.
function in file D:\Aalto\2324\BScThesis\FullRepo\parallelsimulations_finitebath\src\total_hamiltonian.m
Copy to new window for comparing multiple runs

Lines where the most time was spent

Line Number	Code	Calls	Total Time	% Time	Time Plot
42	`g = mutual*(triu(-(gamma/sqrt(...`	25	1.274 s	47.4%	
71	`H = [H1, lambda; lambda', w];`	25	0.914 s	34.0%	
48	`H1 = g+g';`	25	0.372 s	13.9%	
73	`end`	25	0.117 s	4.4%	
52	`H1(1:N+1:end) = omega_j;`	25	0.004 s	0.2%	
All other lines			0.005 s	0.2%	
Totals			2.686 s	100%	

Children (called functions)
No children

Function listing

```
 time    Calls   line
                  30 function H = total_hamiltonian (N, w, mutual, gamma, omega_j)
                  31
                  32 % BATH
                  33
                  34 % Constructs an upper triangular N-by-N matrix of uniformly distributed
                  35 % random numbers between -(gamma/sqrt(N)) and (gamma/sqrt(N)).
                  36 % It represents the normalized by 1/sqrt(N) coupling strength between the
                  37 % spins on the bath, the off-diagonal elements of the Hamiltonian matrix.
                  38 % Thanks to the symmetry of the coupling between two spins and randomness,
                  39 % it is enough to take only the upper triangular matrix.
                  40 % g is relevant only if internal coupling is to be considered for the bath
                  41 % model (mutual=1), otherwise (mutual=0) it becomes zero.
 1.274      25    42 g = mutual*(triu(-(gamma/sqrt(N)) + 2*(gamma/sqrt(N))*rand(N),1));
                  43
                  44 % Constructs a symmetric matrix of coupling strengths, by taking the sum of
                  45 % the upper triangular coupling strength matrix and its transpose. The
                  46 % (Hermitian this way) Hamiltonian is constructed with diagonal elements
                  47 % being zero.
 0.372      25    48 H1 = g+g';
                  49
                  50 % Correct the Hamiltonian by replacing its diagonal elements with the
                  51 % energy gaps of the spins of the bath.
 0.004      25    52 H1(1:N+1:end) = omega_j;
```

```

% Diagonalize the bath Hamiltonian. vek1 is a matrix with column
% eigenvectors and ek1 is a diagonal matrix of eigenvalues.
% [vek1, ek1] = eig(H1);

% BATH AND QUBIT

% Generates a column vector of uniformly distributed random numbers
% between -(gamma/sqrt(N)) and (gamma/sqrt(N)). It represents the
% normalized by 1/sqrt(N) coupling strength between the qubit and the
% spins on the bath.
% That sets the coupling between the spins and the coupling of the qubit
% with the spins at the same level.
0.004   25   lambda = -(gamma/sqrt(N)) + 2*(gamma/sqrt(N))*rand(N,1);

% Build the total Hamiltonian by concatenating lambda and its transpose
% (due to symmetry) as the last (N+1) column and last (N+1) row, while the
% last diagonal element is the frwquency of the qubit.
0.914   25   H = [H1, lambda; lambda', w];

0.117   25   end
```