MMSE FOM: HH Calculation Speed up
COM Commit Request 4p16_2

Adam Gregory, Samtec
Hansel D'Silva, Amphenol

Summary

- Improve COM runtime in MMSE_FOM calculation of HH matrix
- The update speeds up HH calculation by 8x to 10x

Code Change

- The H matrix came from a Toeplitz, so it is just the same column repeated but shifted
- Matlab doesn't inherently realize that multiplying the transpose of such a matrix by itself can be simplified
- The only unique values in the output matrix can be found by multiplying the first column of the matrix by itself
- Then those unique values can be distributed by computing a Toeplitz
  - The lag_idx has been substituted in place of directly calling Toeplitz. This is significantly faster.

```
% OLD
if param.N_bg ~= 0
    H=H( :,[1:Nfix idx+param.RxFFE_cmx+1 ]);
end
HH= H'*H;

% NEW
keep_idx = 1:Nfix;
if param.N_bg ~= 0
    keep_idx = [1:Nfix idx+param.RxFFE_cmx+1];
end
HH_val = transpose(H(:,1))*H;
lag_idx = abs(keep_idx(:) - keep_idx) + 1;
HH = HH_val(lag_idx);
```

Example Form of H matrix

A	0	0	0	0
B	A	0	0	0
C	B	A	0	0
D	C	B	A	0
E	D	C	B	A
F	E	D	C	B
0	F	E	D	C
0	0	F	E	D
0	0	0	F	E
0	0	0	0	F

Example of runtime improvement

- Old code on the left. New code on the right
- Total runtime was 43 seconds vs. 23 seconds

```
                92      % Keep the original form here for documentation
                93      % The form of the speed-up is difficult to tell that it is just H'*H
< 0.01     87802   94    if param.N_bg ~= 0
  2.73     87802   95        H=H( :,[1:Nfix idx+param.RxFFE_cmx+1 ]);
< 0.01     87802   96    end
 18.44     87802   97    HH= H'*H;
< 0.01     87802   98 end
```

```
                61    % keep_idx = indices of main taps and floating taps
  0.03     87802   62    keep_idx = 1:Nfix;
< 0.01     87802   63    if param.N_bg ~= 0
                64        %max_idx = max(idx)+param.RxFFE_cmx+1;
  0.12     87802   65        keep_idx = [1:Nfix idx+param.RxFFE_cmx+1];
< 0.01     87802   66    end
                67
                68    % HH = H'*H but because H is the same shifted column repeated over a
                69    % the unique values of H'*H can be found by multiplying the first co
                70    % HH_val = the column vector containing the N unique values of H'*H,
  1.46     87802   71    HH_val = transpose(H(:,1))*H;
                72
                73    % The final HH is then the toeplitz of HH_val where only the first N
                74    % If there are floating taps, only keep the indices of HH where RxFF
                75    % This block has been replaced by faster block below, but it is kept
                76    %HH = toeplitz(HH_val(1:max_idx));
                77    %if param.N_bg ~= 0
                78    %    HH = HH(keep_idx, keep_idx);
                79    %end
                80
                81    % This block is equivalent to the above commented block using toepli
                82    % Instead of running toeplitz and then sub-indexing it, create the '
                83    % Calling keep_idx(:) - keep_idx is the same as taking the differenc
                84    % Example:
                85    %   A = repmat(keep_idx(:),[1 length(keep_idx)]);
                86    %   B = repmat(keep_idx, [length(keep_idx) 1]);
                87    %   C = A-B = keep_idx(:) - keep_idx
                88    % Avoiding repmat is about 3x faster
  0.16     87802   89    lag_idx = abs(keep_idx(:) - keep_idx) + 1;
  0.55     87802   90    HH = HH_val(lag_idx);
```

Impact on Output

- The new computation of HH has numerical differences < 1e-15 vs. the old HH
- This causes small differences in final COM output
- Showing some typical differences in output here

```
Field "sgm_G" is different
Difference = 4.70544e-17
Field "sgm_rjit" is different
Difference = 6.83047e-18
Field "sgm_N" is different
Difference = 4.67291e-17
Field "sgm_TX" is different
Difference = 1.27936e-17
Field "sgm_isi" is different
Difference = 4.98733e-17
Field "sgm_Q" is different
Difference = 5.85469e-18
Field "sigma_before_clip" is different
Difference = 4.996e-16
Field "peak_clip" is different
Difference = 1.2837e-15
Field "P2ptopsigma_clip" is different
Difference = -3.55271e-15
Field "VEO_mV" is different
Difference = 5.7232e-14
Field "COM_orig" is different
Difference = 3.88023e-14
Field "DER_DFE" is different
Difference = 2.82977e-17
Field "DER_MLSE" is different
Difference = 9.21572e-18
Field "COM_dB" is different
Difference = 3.88578e-14
```

List of CR and KR Channels on 802.3dj Public Area

- Below are the 200 Gb/s CR and KR channel data (TP0 to TP5)- with crosstalk.
  1. akinwale_3dj_01_2311: x4
  2. weaver_3dj_02_2311: x12
  3. akinwale_3dj_01_2310: x7
  4. lim_3dj_07_2309: x1
  5. lim_3dj_04_230629: x1
  6. lim_3dj_03_230629: x1
  7. weaver_3dj_elec_01_230622: x8
  8. shanbhag_3dj_01_2305: x6
  9. shanbhag_3dj_02_2305: x4
  10. kocsis_3dj_02_2305: x5
  11. weaver_3dj_02_2305: x36
  12. mellitz_3dj_02_elec_230504: x27
  13. weaver_3dj_02_2303: x5
  14. mellitz_3dj_02_2303: x54

Table 178–14—Device termination and package model parameters

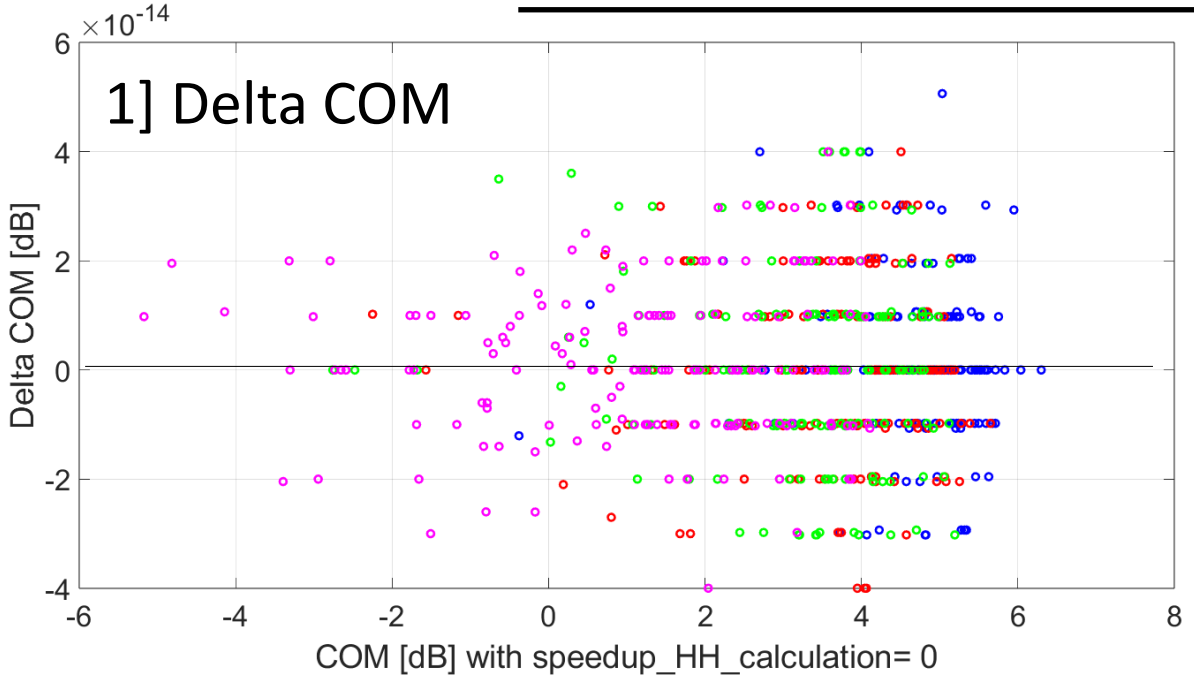
Parameter	Symbol	Value	Units
Device termination model			
Single-ended device capacitance for stage 1	$C_d^{(1)}$	40×10^{-6}	nF
Single-ended device capacitance for stage 2	$C_d^{(2)}$	90×10^{-6}	nF
Single-ended device capacitance for stage 3	$C_d^{(3)}$	110×10^{-6}	nF
Single-ended device series inductance for stage 1	$L_s^{(1)}$	0.13	nH
Single-ended device series inductance for stage 2	$L_s^{(2)}$	0.15	nH
Single-ended device series inductance for stage 3	$L_s^{(3)}$	0.14	nH
Single-ended bump capacitance	C_b	30×10^{-6}	nF
Device package model, class A			
Transmission line parameter γ_0	γ_0	5×10^{-4}	1/mm
Transmission line parameter a_1	a_1	8.9×10^{-4}	$ns^{1/2}/mm$
Transmission line parameter a_2	a_2	2×10^{-4}	ns/mm
Transmission line parameter τ	τ	6.141×10^{-3}	ns/mm
Transmission line 1 length, Test 1	$z_p^{(1)}$	33	mm
Transmission line 1 length, Test 2	$z_p^{(1)}$	12	mm
Transmission line 1 characteristic impedance	$Z_c^{(1)}$	87.5	Ω
Transmission line 2 length	$z_p^{(2)}$	1.8	mm
Transmission line 2 characteristic impedance	$Z_c^{(2)}$	92.5	Ω
Device package model, class B			
Transmission line parameter γ_0	γ_0	5×10^{-4}	1/mm
Transmission line parameter a_1	a_1	6.5×10^{-4}	$ns^{1/2}/mm$
Transmission line parameter a_2	a_2	2.93×10^{-4}	ns/mm
Transmission line parameter τ	τ	6.141×10^{-3}	ns/mm
Transmission line 1 length, Test 1, Tx / Rx	$z_p^{(1)}$	45 / 44	mm
Transmission line 1 length, Test 2, Tx / Rx	$z_p^{(1)}$	30 / 29	mm
Transmission line 1 characteristic impedance	$Z_c^{(1)}$	87.5	Ω
Transmission line 2 length	$z_p^{(2)}$	2	mm
Transmission line 2 characteristic impedance	$Z_c^{(2)}$	95	Ω
Transmission line 3 length	$z_p^{(3)}$	1.3	mm
Transmission line 3 characteristic impedance	$Z_c^{(3)}$	100	Ω
Transmission line 4 length	$z_p^{(4)}$	1.5	mm
Transmission line 4 characteristic impedance	$Z_c^{(4)}$	78	Ω

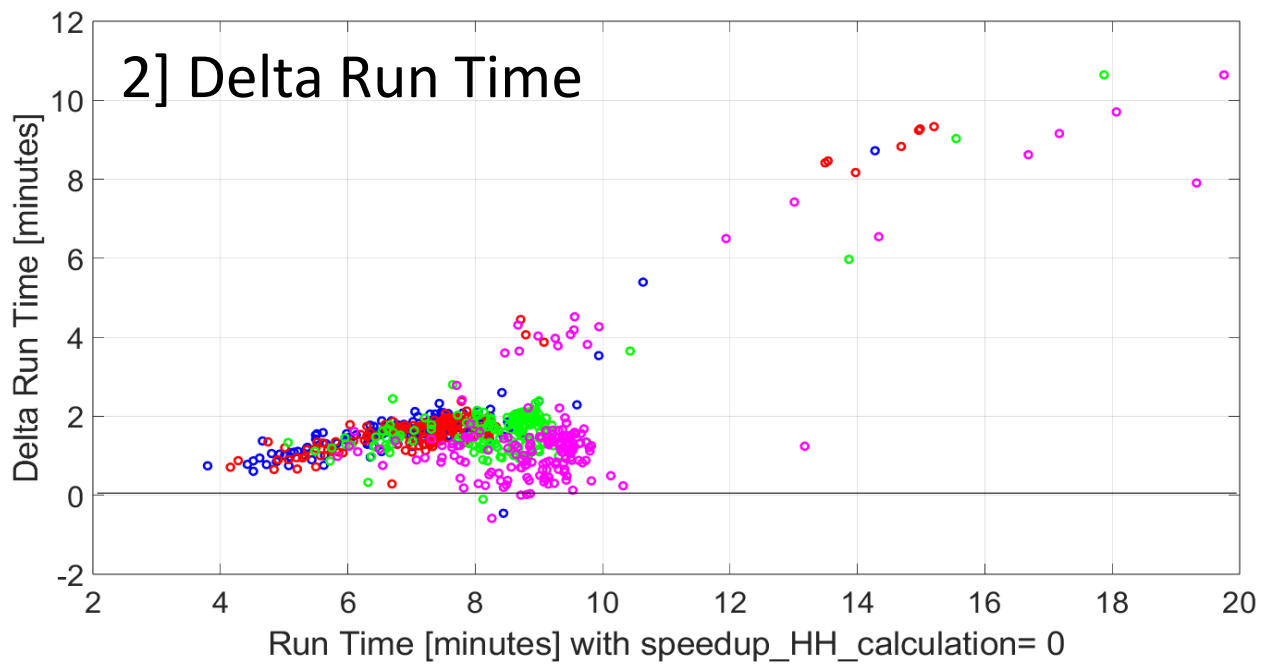
- Total of 171 channels from the IEEE 802.3dj Public Area are evaluated against the class A and class B package models.
  - Total cases= 171x4= 684 cases.

Results

Device pkg model, class A with 12 mm
Device pkg model, class A with 33 mm
Device pkg model, class B with 30/ 29 mm
Device pkg model, class B with 45/ 44 mm

Delta= speedup_HH_calculation= 0 – speedup_HH_calculation= 1





- The new computation of HH has negligible impact (delta of 1e-14) on the value of COM.
- The new computation of HH has significant improvement (as much as 12 minutes) on the value of run time.

Thank You!