

EQ Optimization: Defer calculation of RxFFE floating taps

COM Commit Request 4p16_5

Adam Gregory, Samtec

Summary

- ❑ Most of the run time in COM is spent finding floating RxFFE taps. This is because for every loop in `optimize_fom`, the FOM is found for every possible location of floating RxFFE taps.
- ❑ If the assumption is made that the optimal EQ doesn't really depend on where the floating taps are placed, then EQ optimization can be performed without floating RxFFE taps. Then the floating taps can be found using the best EQ
- ❑ This significantly improves runtime
- ❑ In many cases, the output is the same because the same Optimal EQ is found. In some cases, there is a small change in output.

Parameter Update

- ❑ This feature requires one additional parameter added to the configuration spreadsheet
- ❑ Keyword: **Defer_Floating_RxFFE**
- ❑ Default value = 0
- ❑ If set to 1, the feature is enabled

Code Change

- ❑ This update uses the new architectural update that allows running `optimize_fom` only on the Best EQ setting. The floating taps are temporarily turned off. `Optimize_fom` is run. Then floating taps are turned back on. `Optimize_fom` is run again but only on the best setting.

```
% Check parameter settings that trigger a re-run of optimize_fom with Best EQ
Run_single_optimize = 0;
% Defer Floating RxFFE
restore_floating_rxffe = false;
if param.Defer_Floating_RxFFE
    Run_single_optimize = 1;
    restore_floating_rxffe = true;
end

fom_result = optimize_fom(OP,param, chdata, sigma_bn,do_C2M);
if fom_result.eq_failed ; return; end % RIM 12-20-2023

% If Delay Floating RxFFE is enabled, the optimize_fom sweep ran without floating taps
% Now turn floating taps on and re-run optimize_fom with the best EQ
if restore_floating_rxffe
    param.Floating_RXFFE = true;
    param.N_bg = param.restore_N_bg;
end

% Re-run optimize_fom with Best EQ if any condition has triggered it
if Run_single_optimize
    fom_result = optimize_fom_BestEQ(fom_result, OP, param, chdata, sigma_bn, do_C2M);
end
```

Runtime Improvement

- ❑ Runtime improved from 45 seconds to 11 seconds in this example case
- ❑ Previously 35 seconds were spent in Compute RxFFE. In the update, only 2 seconds are spent in that function
- ❑ Change in COM was 0.014dB

Original Runtime: 45 seconds

com_ieee8023_	1	44.731 s	1.693 s	
optimize_fom	2	35.633 s	0.179 s	
OptFom_Compute_RxFFE	934	35.100 s	0.073 s	
MMSE	934	33.057 s	0.428 s	
FOM_rxffe_floating_taps	934	30.868 s	1.160 s	
MMSE_FOM	325386	29.809 s	29.809 s	

Runtime using Defer Floating Taps: 11 seconds

com_ieee8023_	1	10.828 s	1.626 s	
s21_to_impulse_DC	15	2.389 s	1.707 s	
process_sxp	1	2.308 s	0.022 s	
optimize_fom	2	2.279 s	0.111 s	
get_TDR	8	2.198 s	0.459 s	
OptFom_Compute_RxFFE	935	1.920 s	0.032 s	
conv_fct	8198	1.679 s	1.679 s	
MLSE_U1_c_178A	1	1.634 s	0.098 s	

Fast Runtime Recipe

- ❑ This table gives a recipe on enabling all the fast runtime features
- ❑ This recipe should make TxFFE sweeps plausible

Parameter Name	Default Value	Fast Value	Description
Local Search	0	2	Skip values 2 indices away from best value. Now includes sample phase
Non-zero Local Search Method	0	1	The adaptive Local Search method Hansel added. Able to skip more EQ settings than Local Search alone
Defer_Floating_RxFFE	0	1	Do not apply Floating RxFFE taps during Optimize FOM. Apply after using the Best EQ found in Optimize FOM
Clip Method	Slow	Fast	Fast calculation of Quantization Noise. Set to Slow after Optimize FOM, so now similar to using Slow

Thank You!