LAMPIRAN A

Simulasi1.m

```
Lws = [3:1:50]-1;
nu = 32;
Nlw = length(Lws);
Ncsa = 8;
Rmel = zeros(Nlw,Ncsa);
Runt = zeros(Nlw,Ncsa);
hh = waitbar(0,'Melakukan Perhitungan...');
for j=1:Ncsa
  j
  eval(['load ../channels/csaloop',num2str(j),'.time;']);
  eval(['h = (csaloop',num2str(j),'(:,2));']);
  h = h/norm(h);
  Lh = length(h)-1;
  for i=1:Nlw
    Lw = Lws(i);
    Lc = Lw+Lh;
    [wmelsa,dopt_mel] = mssnr(h,nu+1,Lw+1,0,Lc-nu);
    delta = dopt_mel;
    H = convmtx(h,Lw+1);
    Hwin = H([delta+1:delta+nu+1],:);
    Hwall = H([1:delta,delta+nu+2:end],:);
    A = Hwall'*Hwall;
    B = Hwin'*Hwin;
    [V,D] = eig(A);
    [dummy,index] = min(abs(diag(D)));
    w_unt = V(:,index);
    [w_unt_s,w_unt_a] = symasym(w_unt);
```

```
    [w_mel_s,w_mel_a] = symasym(wmelsa);


        Rmel(i,j) = (norm(w_mel_a)/norm(w_mel_s))^2;
    Runt(i,j) = (norm(w_unt_a)/norm(w_unt_s))^2;
    waitbar( ((j-1)*Nlw+i)/(Ncsa*Nlw), hh);
  end
end
close(hh);
aRmel = mean(Rmel,2);
aRunt = mean(Runt,2);
figure(1);
clf; set(0,'DefaultAxesFontSize',16);
plot(Lws+1,aRunt,'r-');
xlabel('length of TEQ','FontSize',16);
ylabel('||w_{skew}||^2/||w_{sym}||^2','FontSize',16);
axis([0 240 0 0.15]);
grid on;
figure(2);
clf; set(0,'DefaultAxesFontSize',16);
plot(Lws+1,aRmel,'b-', Lws+1,aRunt,'r--');
xlabel('Pnjang TEQ','FontSize',16);
ylabel('||w_{skew}||^2/||w_{sym}||^2','FontSize',16);
legend('MSSNR','MSSNR-UNT');
grid on;
```

mssnr.m

```
function [wopt,dopt] = mssnr(h,Nb,tLw,dmin,dmax)
h = h(:);
```

```
lambdaopt = 0;
H = convmtx(h,tLw);
C = H'*H;
for delay = dmin:dmax
  Hwin = H(delay+1:delay+Nb,:);
  B = Hwin'*Hwin;
  [evecs,evals] = eig(B,C);
  [lambda,ind] = max(diag(evals));
  if lambda > lambdaopt
    wopt = evecs(:,ind);
    dopt = delay;
    lambdaopt = lambda;
  end
end
```

symasym.m

```
function [ws,wa] = symasym(w)
Lw = length(w);
meas = zeros(2*Lw-1,1);
for center = 1:Lw
  left = center-1;
  right = Lw-center;
  half = min(left,right);
  wmid = w(center-half:center+half);
  ws = zeros(Lw,1);
  ws(center-half:center+half) = (wmid+flipud(wmid))/2;
  wa = w-ws;
  meas(center) = norm(ws)/norm(wa);
```

```
end
for center = Lw+1:2*Lw-1
  left = center-Lw;
  right = Lw-left;
  half = min(left,right);
  wmid = w(left-half+1:left+half);
  ws = zeros(Lw,1);
  ws(left-half+1:left+half) = (wmid+flipud(wmid))/2;
  wa = w-ws;
  meas(center) = norm(ws)/norm(wa);
end
[dummy,center] = max(meas);
if center <= Lw
  left = center-1;
  right = Lw-center;
  half = min(left,right);
  wmid = w(center-half:center+half);
  ws = zeros(Lw,1);
  ws(center-half:center+half) = (wmid+flipud(wmid))/2;
  wa = w-ws;
else
  left = center-Lw;
  right = Lw-left;
  half = min(left,right);
  wmid = w(left-half+1:left+half);
  ws = zeros(Lw,1);
  ws(left-half+1:left+half) = (wmid+flipud(wmid))/2;
  wa = w-ws;
```

```
end
```

Simulasi2.m

```
% parameter sistem
N  = 512;            % FFT size
nu = 32;             % CP length
M  = N + nu;         % panjang simbol keseluruhan
T  = 16;             % equalizer length
load ../channels/csaloop4.time
h = (csaloop4(:,2));
h = h/norm(h);
L = length(h);       % Panjang channel
sx = 1;              % sigma_x
ssx = sx*sx;
SNRs = 40;           % dB
% parameter simulasi
iter = 1500;         % jumlah symbols
mu0 = 0.25;
starttime = 2;
% solusi optimal Melsa
[wtb,dopt] = mssnr(h,nu,T,0,150);
ctb = conv(wtb,h);
[SSNRtb,dummy] = Jssnr(ctb,nu,0);
[dummy, index] = max(abs(h));
index = index-1;      % (channel delay)
delta = dopt;

% equalizer delay untuk inisialisasi
```

```
deq = 7;
% equalizer initialization
w = zeros(T,iter);
winit = zeros(T,1);
winit(1+deq) = 1;
winit = winit + flipud(winit);
SSNR = zeros(1,iter);
Jm = zeros(1,iter);
b = zeros(1,iter);
Jmerry_final = zeros(1,length(SNRs));
Jmelsa_final = zeros(1,length(SNRs));
BRmerry_final = zeros(1,length(SNRs));
BRmelsa_final = zeros(1,length(SNRs));
for SNRind = 1:length(SNRs)
%-------------------------------------------------
SNR = SNRs(SNRind);
g = 10.^(-SNR/10);
sn = sx*sqrt(g);
ssn = sn*sn;
Rn = ssn*eye(T);
Jmtb = Jmerry(ctb,wtb,nu,dopt,ssx,Rn);
btb = sum( bitrate(ctb,wtb,nu,dopt,ssx,ssn) );
% solusi optimal Merry
[wopt,SSNRopt] = merryopt(h,nu-1,T,dopt+1,g);
copt = conv(wopt,h);
[SSNRopt,dummy] = Jssnr(copt,nu,0);
copt = conv(wopt,h);
Jmopt = Jmerry(copt,wopt,nu,dopt,ssx,Rn);
```

```
bopt = sum( bitrate(copt,wopt,nu,dopt,ssx,ssn) );
r = adslout(h,sx,sn,iter);
% inisialisasi equalizer
w(:,starttime) = winit;
cstart = conv(w(:,starttime),h);
mu = mu0;
hwin = waitbar(0,'Menjalankan MERRY...');
for k=starttime:iter-1
  c = (conv( w(:,k), h));
  [SSNR(k),dummy] = Jssnr(c,nu,0);
  Jm(k) = Jmerry(c,w(:,k),nu,dopt,ssx,Rn);
  b(k) = sum( bitrate(c,w(:,k),nu,dopt,ssx,ssn) );
  % update w
  i1 = M*(k-1)+nu+delta;
  i2 = i1+N;
  v = w(1:T/2,k);
  rtilde = ( r(i1:-1:i1-(T/2)+1) + r(i1-T+1:i1-(T/2)) )' - ...
        ( r(i2:-1:i2-(T/2)+1) + r(i2-T+1:i2-(T/2)) )';
  e = v' * rtilde;
  v = v - mu * e * rtilde;
  w(:,k+1) = [v; flipud(v)];
  % normalisasi
  w(:,k+1) = w(:,k+1)/norm(w(:,k+1));
  waitbar(k/iter,hwin);
end
close(hwin);
cfinal = conv( w(:,k+1), h);
[SSNR(k+1),dummy] = Jssnr(cfinal,nu,0);
```

```
Jm(k+1) = Jmerry(cfinal,w(:,k+1),nu,dopt,ssx,Rn);
b(k+1) = sum( bitrate(cfinal,w(:,k+1),nu,dopt,ssx,ssn) );
Jmerry_final(SNRind) = Jm(k+1);
Jmelsa_final(SNRind) = Jmtb;
BRmerry_final(SNRind) = b(k+1);
BRmelsa_final(SNRind) = btb;
%-----------------------------------------------
end
Background = .8*ones(1,3);
figure(1);
clf
set(0,'DefaultAxesFontSize',14);
%
subplot(4,1,1);
plot([0:T+L-2],cstart/max(abs(cstart)),'r-');
title('Channel-Equalizer Combinations','FontSize',16);
ylabel('initial','FontSize',16);
v = axis;
xdata = [delta delta+nu-1 delta+nu-1 delta];
ydata = [v(3) v(3) v(4) v(4)];
patch('Xdata',xdata,'Ydata',ydata,...
    'EraseMode','xor','EdgeColor','w','FaceColor',Background);
axis([0 200 v(3) v(4)]);
%
subplot(4,1,2);
plot([0:T+L-2],cfinal/max(abs(cfinal)),'r-');
ylabel('final','FontSize',16);
v = axis;
```

```
xdata = [delta delta+nu-1 delta+nu-1 delta];
ydata = [v(3) v(3) v(4) v(4)];
patch('Xdata',xdata,'Ydata',ydata,...
    'EraseMode','xor','EdgeColor','w','FaceColor',Background);
axis([0 200 v(3) v(4)]);
%
subplot(4,1,3);
plot([0:T+L-2],copt/max(abs(copt)),'r-');
ylabel('Merry Opt.','FontSize',16);
v = axis;
xdata = [delta delta+nu-1 delta+nu-1 delta];
ydata = [v(3) v(3) v(4) v(4)];
patch('Xdata',xdata,'Ydata',ydata,...
    'EraseMode','xor','EdgeColor','w','FaceColor',Background);
axis([0 200 v(3) v(4)]);
%
subplot(4,1,4);
plot([0:T+L-2],ctb/max(abs(ctb)),'r-');
ylabel('Melsa','FontSize',16);
v = axis;
xdata = [delta delta+nu-1 delta+nu-1 delta];
ydata = [v(3) v(3) v(4) v(4)];
patch('Xdata',xdata,'Ydata',ydata,...
    'EraseMode','xor','EdgeColor','w','FaceColor',Background);
axis([0 200 v(3) v(4)]);
xlabel('tap index','FontSize',14);

figure(2);
```

```
clf
set(0,'DefaultAxesFontSize',14);
subplot(2,1,1);
plot([0:T+L-2],cstart/max(abs(cstart)),'r-');
title('Channel-Equalizer Combinations','FontSize',16);
ylabel('initial','FontSize',16);
v = axis;
v = [v(1) v(2) -0.2 1];
axis([0 200 v(3) v(4)]);
hold on;
plot([delta delta],[v(3) v(4)],'k--',...
  [delta+nu-1 delta+nu-1],[v(3) v(4)],'k--',...
  [0 200],[0 0],'k-');
hold off;
%
subplot(2,1,2);
plot([0:T+L-2],cfinal/max(abs(cfinal)),'r-');
ylabel('final','FontSize',16);
v = axis;
axis([0 200 v(3) v(4)]);
hold on; plot([delta delta],[v(3) v(4)],'k--',...
  [delta+nu-1 delta+nu-1],[v(3) v(4)],'k--',...
  [0 200],[0 0],'k-');
hold off;
xlabel('tap index','FontSize',16);

Tsym = 246.4e-6;
figure(3);
```

```
clf
set(0,'DefaultAxesFontSize',9);
subplot(2,1,1);
semilogy([starttime:iter],Jm(starttime:iter),'b-',...
    [starttime,iter],Jmopt*[1,1],'k--','LineWidth',2);
ylabel('MERRY cost','FontSize',14);
legend('Adapted','Optimal MERRY',1);
grid on
axis([0 iter 0.85e-4 1]);
%
subplot(2,1,2);
plot([starttime:iter],b(starttime:iter)/Tsym,'b-',...
    [starttime,iter],bopt/Tsym*[1,1],'k--',...
    [starttime,iter],btb/Tsym*[1,1],'k:','LineWidth',2);
ylabel('bit rate (bps)','FontSize',14);
xlabel('indeks simbol','FontSize',14);
legend('Adapted','Optimal MERRY','Max SSNR',4);
grid on
```

Simulasi3.m

```
nu = 32;
Nb = nu+1;
Nws = [3:128];
numNws = length(Nws);
SNRdb = 60;
SNRlinear = 10^(SNRdb/10);
ssx = 1;
ssn = ssx/SNRlinear;
```

```
MMSEopt = zeros(numNws,8);
MMSEsym = zeros(numNws,8);
BRopt = zeros(numNws,8);
BRsym = zeros(numNws,8);
hh = waitbar(0,'Iterasi CSA loops dan panjang TEQ');
for csanum=1:8
eval(['load ../channels/csaloop',num2str(csanum),'.time']);
eval(['h = csaloop',num2str(csanum),'(:,2);']);
h = h/norm(h);
Lh = length(h);
Dmin = 0;
Dmax_temp = Lh-nu-1;
for iNw = 1:numNws
  Nw = Nws(iNw);
  Dmax = Dmax_temp+(Nw-1);
  % optimal MMSE TEQ design
  [wopt,bopt,dopt,MMSEopt(iNw,csanum)] = ...
    mmse_unc(h,Nb,Nw,Dmin,Dmax,ssx,ssn);
  copt = conv(h,wopt);
  % Desain MMSE TEQ untuk symmetric TIR
  [wsym,bsym,dsym,MMSEsym(iNw,csanum)] = ...
    mmse_sym(h,Nb,Nw,Dmin,Dmax,ssx,ssn);
  csym = conv(h,wsym);
  % compute bit rates
  BRopt(iNw,csanum) = sum( bitrate(copt,wopt,nu,dopt,ssx,ssn) );
  BRsym(iNw,csanum) = sum( bitrate(csym,wsym,nu,dsym,ssx,ssn) );
  waitbar( (csanum-1)/8 + (iNw/numNws)/8, hh);
end
```

```
end
close(hh)
Nws(18)
rel = (BRsym(18,:) ./ BRopt(18,:)) * 100;
[BRopt(18,:).'/1e6, BRsym(18,:).'/1e6, rel.']
Tsym = 246.4e-6;
BRopt = BRopt/Tsym;
BRsym = BRsym/Tsym;
BRopt_avg = mean(BRopt,2);
BRsym_avg = mean(BRsym,2);
figure(1);
plot([0:length(copt)-1],copt,'b-', [dopt:dopt+nu],bopt,'rx');
title('MMSE design');
v = axis;
axis([0 100 v(3) v(4)]);
figure(2);
plot([0:length(csym)-1],csym,'b-', [dsym:dsym+nu],bsym,'rx');
title('Desain MMSE dengan Symmetric TIR');
v = axis;
axis([0 100 v(3) v(4)]);
figure(3);
semilogy(Nws,MMSEopt,'b-', Nws,MMSEsym,'r--');
title('MMSE');
figure(4);
clf;
set(0,'DefaultAxesFontSize',14);
subplot(2,1,1);
plot(Nws,BRopt(:,1),'b-', Nws,BRsym(:,1),'r--');
```

```
ylabel('bit rate (loop 1)','FontSize',14);
grid on;
subplot(2,1,2);
plot(Nws,BRopt(:,3),'b-', Nws,BRsym(:,3),'r--');
legend('MMSE','Sym-MMSE',4);
xlabel('Panjang TEQ','FontSize',14);
ylabel('bit rate (loop 3)','FontSize',14);
grid on;
figure(5);
clf;
set(0,'DefaultAxesFontSize',14);
plot(Nws,BRopt_avg/1e6,'b-', Nws,BRsym_avg/1e6,'r--');
v = axis;
axis([1 Nws(end) v(3) v(4)]);
legend('MMSE','Sym-MMSE',4);
xlabel('Panjang TEQ','FontSize',14);
ylabel('bit rate (rata-rata), Mbps','FontSize',14);
grid on;
```