

```

%
% Written by:
% --
% John L. Weatherwax          2007-12-07
%
% email: wax@alum.mit.edu
%
% Please send comments and especially bug reports to the
% above email address.
%
%-----

% the number of experiments:
N_EXPS = 100;
%N_EXPS = 1000;

% the number of learning episodes (for complete learning this would need to be much
larger):
nEpisodes = 20;

% the number of random walk states (not including the two terminal states):
%n_rw_states = 17;
n_rw_states = 19;
%n_rw_states = 7;

% the true solution:
% (when zero is the reward for going off the left end):
%Vtruth = (0:(n_rw_states+1))/(n_rw_states+1); Vtruth(1)=[]; Vtruth(end)=[];
% (when -1 is the reward for going off the left end):
Vtruth = -1 + 2*(0:(n_rw_states+1))/(n_rw_states+1); Vtruth(1)=[]; Vtruth(end)=[];

% a vector of learning parameters with 0.0 and 1.0
%alphaV = linspace( 0, 1, 10 );
%alphaV = linspace( 0, 1, 20 );
alphaV = linspace( 0, 1, 100 );

% a vector of TD(lambda) lambdas:
lamVect = [ 0, 0.2, 0.4, 0.6, 0.8, 0.9, 0.95, 0.975, 0.99, 1.0 ];

results_acc=zeros(1,length(lamVect));
results_rep=zeros(1,length(lamVect));
for ni=1:length(lamVect),
    lam=lamVect(ni);
    fprintf('working on lambda=%f...\n',lam);

    tr = zeros(1,N_EXPS);
    fprintf('working on experiment number (of %d): ',N_EXPS);
    for ei=1:N_EXPS,
        if (mod(ei,10)==0 ) fprintf('%d, ',ei); end
        tra = zeros(1,length(alphaV));
        for ai=1:length(alphaV),
            a = alphaV(ai);
            %fprintf('working on alpha=%f...\n',a);
            V_w_et      = rw_online_w_et( lam, a, n_rw_states, nEpisodes );
            tra_acc(ai) = sqrt( mean( (V_w_et-Vtruth).^2 ) );
            V_w_rp      = rw_online_w_replacing_traces( lam, a, n_rw_states, nEpisodes );
            tra_rep(ai) = sqrt( mean( (V_w_rp-Vtruth).^2 ) );
        end
        [be_acc,mai_acc] = min( tra_acc ); % <- get the best alpha
    end
end

```

```
tr_acc(ei) = be_acc;
[be_rep,mai_rep] = min( tra_rep ); % <- get the best alpha
tr_rep(ei) = be_rep;
end
fprintf('\n');
results_acc(ni) = mean( tr_acc );
results_rep(ni) = mean( tr_rep );
end

figure; hold on;
ph1=plot( lamVect, results_acc, '-xr' ); hold on;
ph2=plot( lamVect, results_rep, '-og' ); grid on;
xlabel( 'lambda' ); ylabel( ['average rms error from ',num2str(nEpisodes), ' episodes']
);
legend( [ph1,ph2], {'Accumulating Traces','Replacing Traces'}, 'Location', 'Best' );
%axis( [ 0.0, lamVect(end), 0.05, 0.6 ] );

saveas(gcf,['./rw_acc_vs_rep_',num2str(N_EXPS)],'png');

return;
```