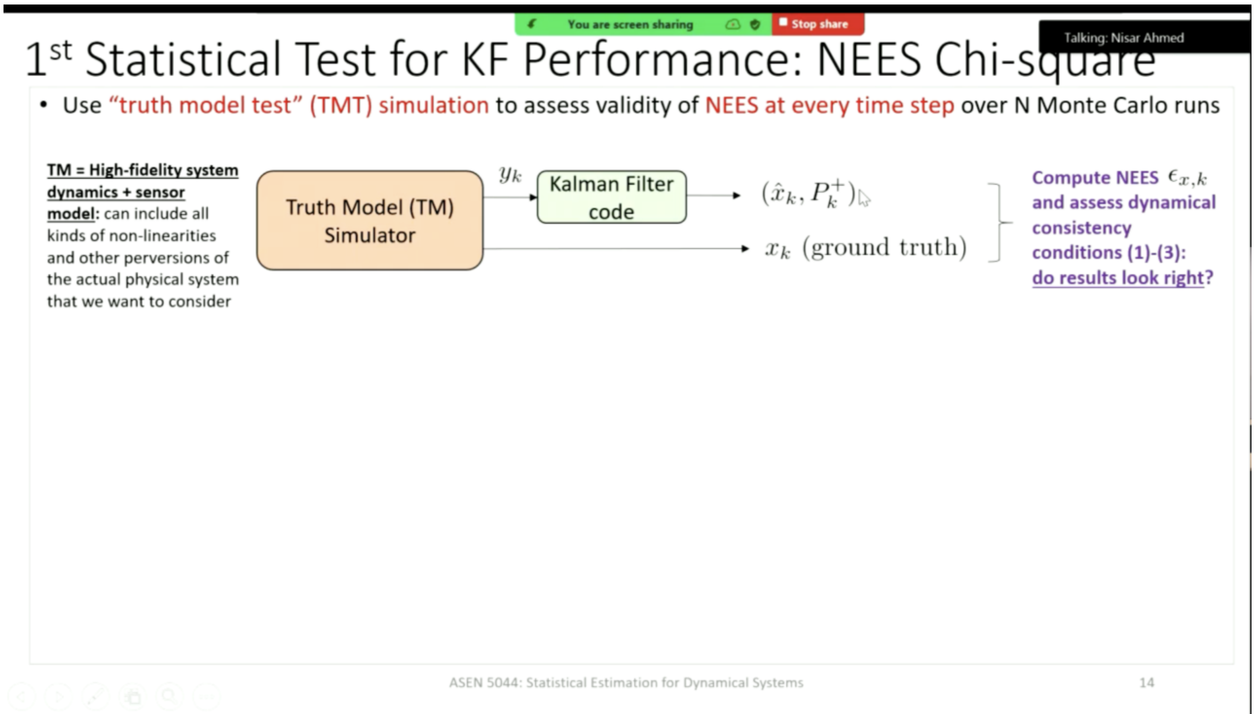
NEES/NIS Design Notes

NEES/NIS design based on model provided in class.



From what I can gather TM is the ode45 work we have done. And Kalman Filter code is the Part4Wrapper.m. Do we just want to integrate NEES/NIS testing into Wrapper or make it a functional call?

$$\epsilon_{x,k} = e_{x,k}^T (P_k^+)^{-1} e_{x,k} \quad \rightarrow \textbf{Normalized estimation error squared (NEES)} \text{ at time k}$$

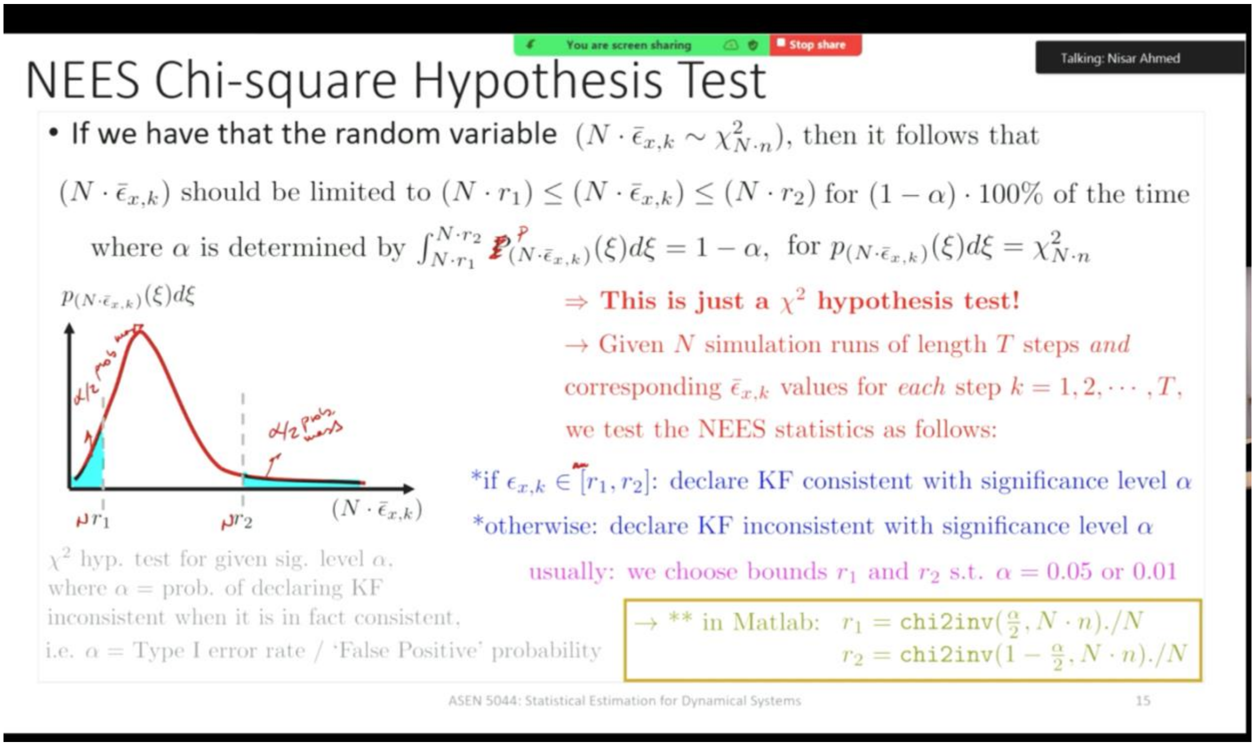
$$\epsilon_{y,k} = e_{y,k}^T (S_k)^{-1} e_{y,k} \quad \rightarrow \textbf{Normalized innovation squared (NIS)} \text{ at time k}$$

Code for ex,k and ey,k:

```
%
% (NEES) error calculated here, that is,
%
%  e_x,k = (xhat - xtrue)'*P^+_k*(xhat - xtrue)
%
%
NEESsshist(k) = (xk_truehist(:,k) - mkp1_plus)'*invPkp1*(xk_truehist(:,k) - mkp1_plus);

%
% (NIS) error calculated here, that is,
%
%  e_y,k = (y - yhat)'*R^(-1)_k*(y - yhat)
%
NISsshist(k) = innov_kp1'*invPyykp1*innov_kp1;
```

Setting our bounds in the Chi-square distro is done in MATLAB with chi2inv, see slide below.



In Code:

```
%
%  N = number of simulation runs
%  n = total number of time step's k
%
% r1x and r2x are the lower and upper bounds of the confidence interval for the NEES statistic

% Compute the confidence intervals for the NEES and NIS statistics
r1x = chi2inv(alphaNEES/2, numberOfSimulationRuns*timeStepk) ./ numberOfSimulationRuns;
r2x = chi2inv(1 - alphaNEES/2, numberOfSimulationRuns*timeStepk) ./ numberOfSimulationRuns;

% r1y and r2y are the lower and upper bounds of the confidence interval for the NIS statistic
r1y = chi2inv(alphaNIS/2, numberOfSimulationRuns*timeStepk) ./ numberOfSimulationRuns;
r2y = chi2inv(1 - alphaNIS/2, numberOfSimulationRuns*timeStepk) ./ numberOfSimulationRuns;
```

Calculating mean for epsilon vectors:nis

; $\bar{\epsilon}_{x,k}$ values for *each* step $k = 1, 2, \cdots, T$,

In code:

```
% Compute the mean and variance of the NEES and NIS samples
NEESmean = mean(NEESsamps, 1);
NISmean = mean(NISsamps, 1);
```

Consistency filter should look like the results on this slide: