Application of Kalman Filtering Technique for SAR Processing of Sparse Satellite Clusters

Subhash Gullapalli

M.S. Thesis Defense

Date: Dec 18th 2002

Committee

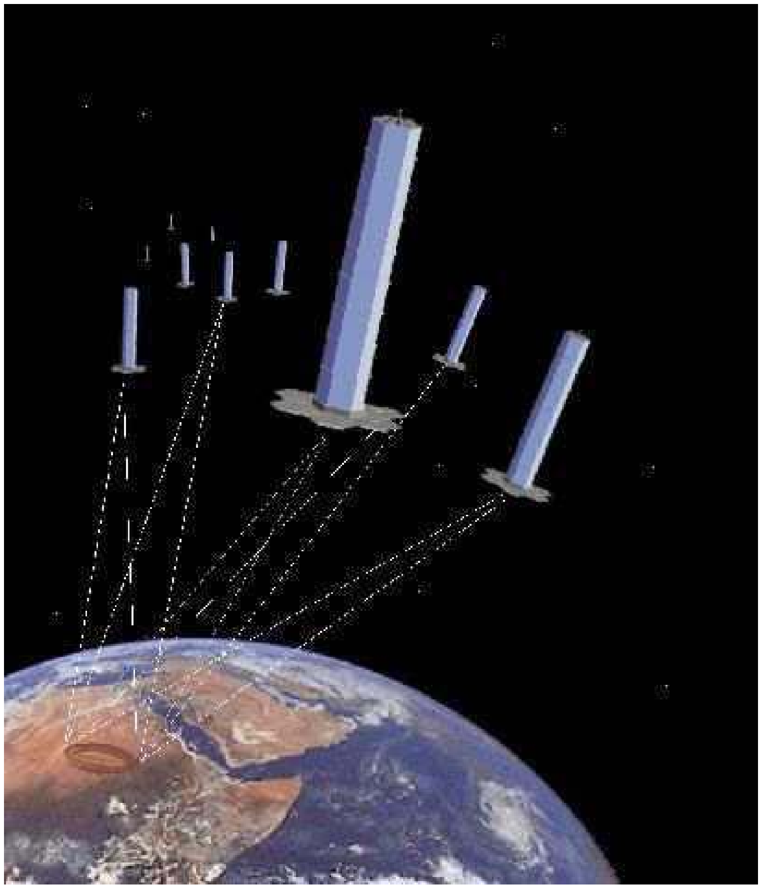
Dr. James Stiles (Chair)

Dr. John Gauch

Dr. Christopher Allen

Information and Telecommunication Technology Center

Sparse Satellite Clusters



- Advantages of Space borne Radar
- Size-Weight-Power Tradeoff
- Ambiguity and Resolution
- Cluster of Satellites called Constellation
- Advantages of Multiple Satellites
- Sparsely Populated Multiple Aperture Spaceborne Radar

Earlier Proposed Filters

- Matched or Correlation Filter
  - Estimator Maximizes signal to noise
  - Unable to Minimize error due to clutter
- Maximum Likelihood Estimator
  - Able to Minimize error due to clutter
  - Unable to minimize error due to noise
- Minimum Mean Square Estimator (MMSE)
  - Reduces Interference (Clutter +Noise)
  - Optimal estimate
  - Computational load

Research Question ?

Is it possible to develop an Estimator able to estimate as accurately as the MMSE, while being able to reduce the computational load inherent in MMSE ?

Overview

- Signal Space Representation of the Radar System
- MMSE Filter
- Problems
- Kalman Filter Implementation (KF)
- Convergence of the KF for different initial conditions
- Reducing the Rank of the error covariance matrix
- Implementation of Kalman Filtering on parallel processors
- Conclusions
- Future Work

Signal Space Representation of the Radar System

$$r(\bar{x}_r,t) = \int_A \gamma_0(\bar{x}) \int_T h(\bar{x}_r, x, t, t')\, s(t')\,dt'\,dA + n(\bar{x}_r,t)$$
$$= \sum_i \gamma_0(\bar{x}_i)\,\rho(\bar{x}_r,\bar{x}_i,t)\Delta A + n(\bar{x}_r,t)$$

$$\mathbf{r} = \mathbf{P}\boldsymbol{\gamma} + \mathbf{n}$$

$\mathbf{r}$ = Entire set of Measurements
$\mathbf{P}$ = Matrix of Normalized Response vectors
γ = Scattering Coefficients Vector
$\mathbf{n}$ = Measurement Noise

- The **received** signal for the Radar Constellation can be **modeled** mathematically.
- This response depends on radar system parameters, propagation, **scattering** characteristics of the surface.
- Signals can be approximately represented by sampled data and interpolation filters can be used to reconstruct the signal
- The radar response model can thus be represented with **vector-matrix** relations.

Information and Telecommunication Technology Center

The MMSE Estimator

$$\hat{\gamma}_i = \mathbf{w}_i^{\mathbf{H}}\mathbf{r}$$

$$\hat{\boldsymbol{\gamma}} = \mathbf{W}\mathbf{r}$$

$$\mathbf{W}_{\mathbf{MMSE}} = \mathbf{K}_{\gamma}\mathbf{P}^{\mathbf{H}}\left[\mathbf{P}\,\mathbf{K}_{\gamma}\mathbf{P}^{\mathbf{H}} + \mathbf{K}_{n}\right]^{\sim 1}$$

$\mathbf{K}_{\gamma} = E\{\gamma\gamma'\}$ is the error correlation matrix

K_n is the noise covariance matrix

$(\cdot)^{H}$ is the conjugate transpose

- SAR processing estimates the scattering from each resolution cell and involves with finding the optimal estimator matrix w
- *A priori* estimates of SNR can be used to construct the **MMSE** filter.
- Weight vector is a **compromise** between noise and clutter.
- Maximizes signal to **interference** (clutter +noise).
- Result is **minimum** estimation error.
- Huge computational load involved due to inverse

Problems

- Problem 1
  How to divide the single large operation into multiple smaller operations?

- Problem2
  How to reduce the computations defining error covariance matrix?

- Problem3
  How to reduce the processing time for matrix multiplications?

Kalman Filtering

General Equations

State Equation

$$\mathbf{y}(l) = \mathbf{A}(l)\,\mathbf{y}(l-1) + \mathbf{w}(l)$$

Measurement Equation

$$\mathbf{z}(l) = \mathbf{H}(l)\,\mathbf{y}(l) + \mathbf{v}(l)$$

$\mathbf{y}(l) =$ Signal Vector

$\mathbf{A}(l) =$ State Transition Matrix

$\mathbf{w}(l) =$ Process Noise

$\mathbf{z}(l) =$ Measurement Vector

$\mathbf{H}(l) =$ Matrix of Constants

$\mathbf{v}(l) =$ Measurement Noise

Specific Equations

$$\gamma(l) = \mathbf{A}(l)\,\gamma(l-1) + \mathbf{u}(l)$$

$$\mathbf{r}(l) = \mathbf{P}(l)\,\gamma(l-1) + \mathbf{n}(l)$$

$\mathbf{A}(l)$ is taken to be an identity matrix

Generally assumed that scattering coefficients are approximately constant with respect to time, space and frequency

Radar Measurements are segmented

$$\gamma(l) = \gamma(l-1) + \mathbf{u}(l)$$

Kalman Filter Implementation

Measurement vector **r** is divided into
L segments. Image estimate is determined as :

$$\hat{\gamma}(l/l) = \gamma(l-1/l-1) + \mathbf{G}(l)\nu(l)$$

from innovation ν :

$$\nu(l) = \mathbf{r}(l) - \mathbf{P}(l)\hat{\gamma}(l-1/l-1)$$

from Kalman Gain **G** :

$$\mathbf{G}(l) = \mathbf{K}_\gamma(l/l-1)\mathbf{P}(l)^H\left[\mathbf{P}(l)\mathbf{K}_\gamma(l/l-1)\mathbf{P}(l)^H + \mathbf{K}_n(l)\right]^{-1}$$

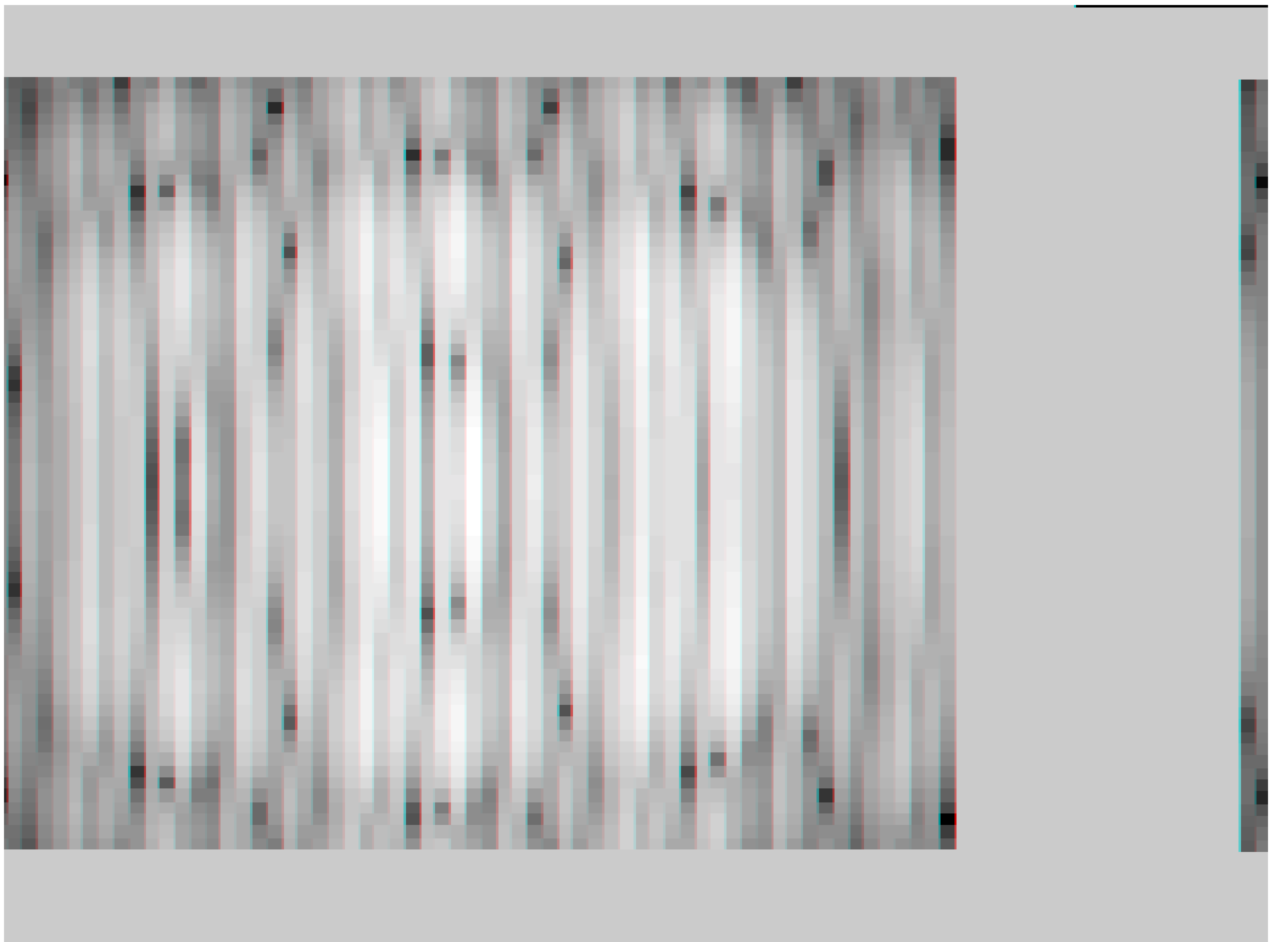
Update Error Covariance

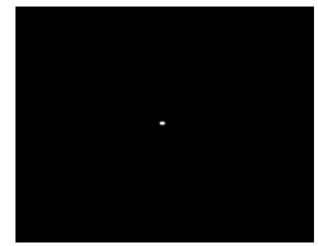
$$\mathbf{K}_\gamma(l/l) = [\mathbf{I} - \mathbf{G}(l)\mathbf{P}(l)]\mathbf{K}_\gamma(l/l-1)$$

and applied to next iteration

- Iterative implementation of MSME
- Process noise is neglected
- Innovation is the new information available in the latest measurement
- Kalman Gain is computed so as to minimize the MSE, and is based on orthogonal principle
- Initially, measurement error due to ambiguities and clutter dominates. in the final stages noise dominates

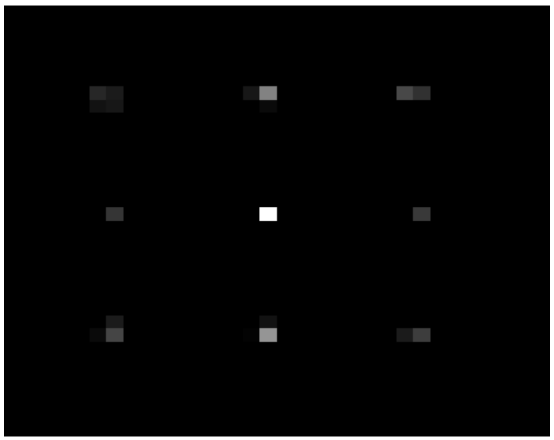
Kalman Filtering Process for a single Target



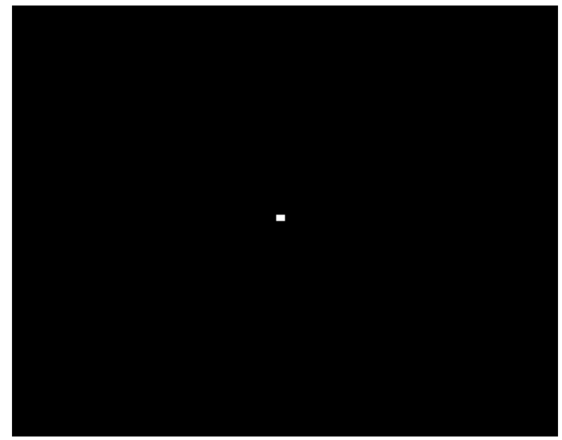


Single Point Target
Resolution 64 by 64

Matched Vs. Kalman filter



Matched Filter Estimate

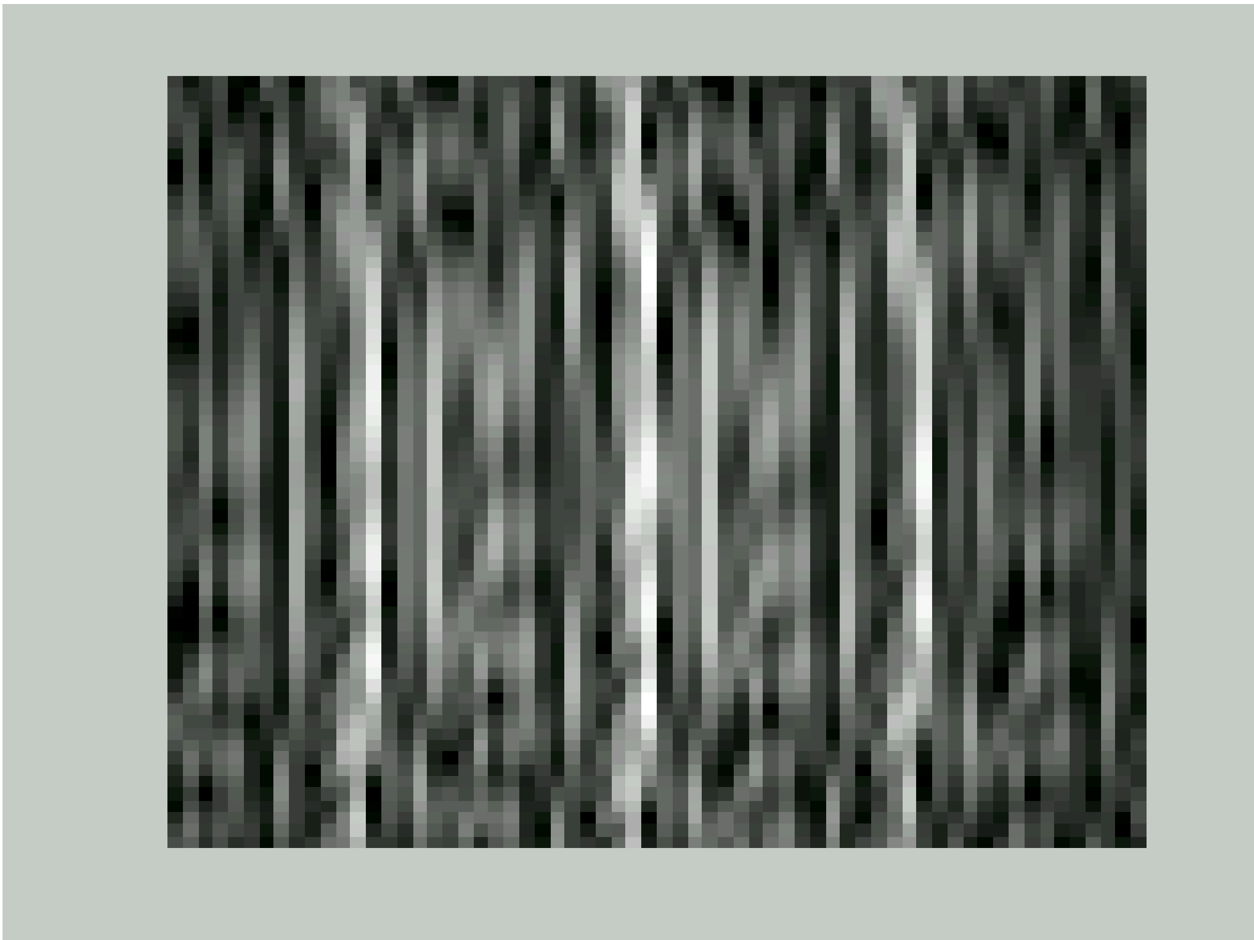


Kalman Filter Estimate

Single Point Target

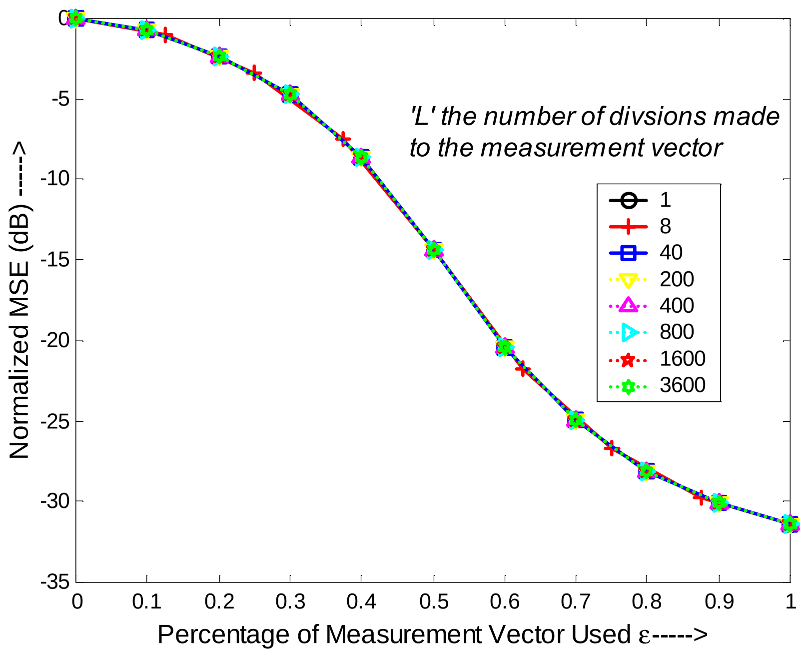
- Matched Filter unable to resolve range and Doppler Ambiguities
- Kalman Filter gives Optimal Estimate

Kalman Filtering Process for a Real Scenario



General SAR Image

Accuracy of Estimate vs. L



Estimation error vs. the percentage of measurement vector processed

$\varepsilon = \frac{l}{L}$, l is the number of iterations done

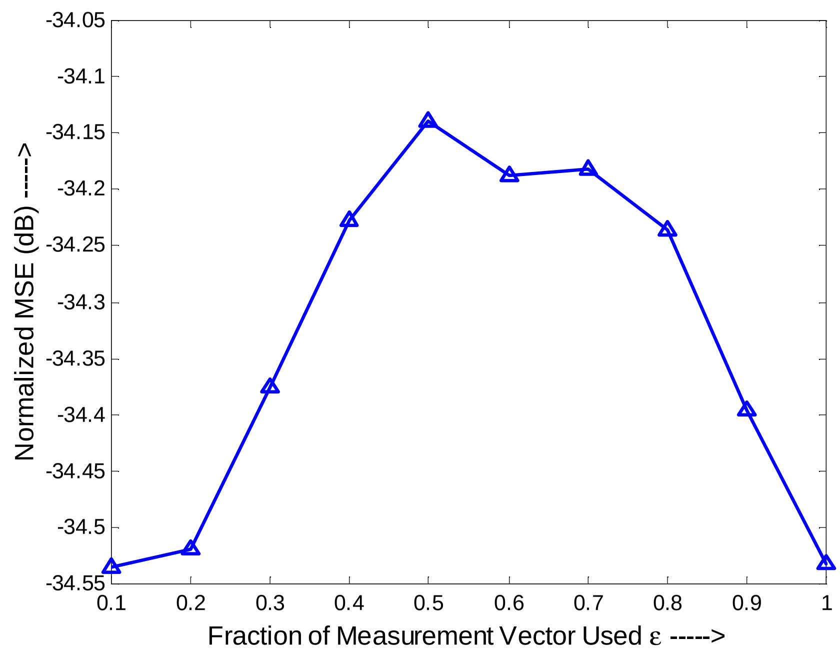
$\mathbf{K}_\gamma(0) = \sigma_\gamma^2 \mathbf{I}$

σ_γ^2 is the expected value of $|\gamma|^2$

- Accuracy of estimate same as MMSE
- Why is there no improved accuracy?
- Assumption: Scattering Coefficients are approximately constant with time.
- Difficult to simulate this variation
- In real time scenarios, there will be improvement if modeled optimally

Information and Telecommunication Technology Center

Perfect Initial Conditions



Estimation error vs. the percentage of measurement vector processed

- What if the initial conditions are perfect?
- Final Error Covariance and estimate are saved and taken as the initial values
- There isn't any improvement! Why ?
- No Innovation Energy left in the measurements

Information and Telecommunication Technology Center

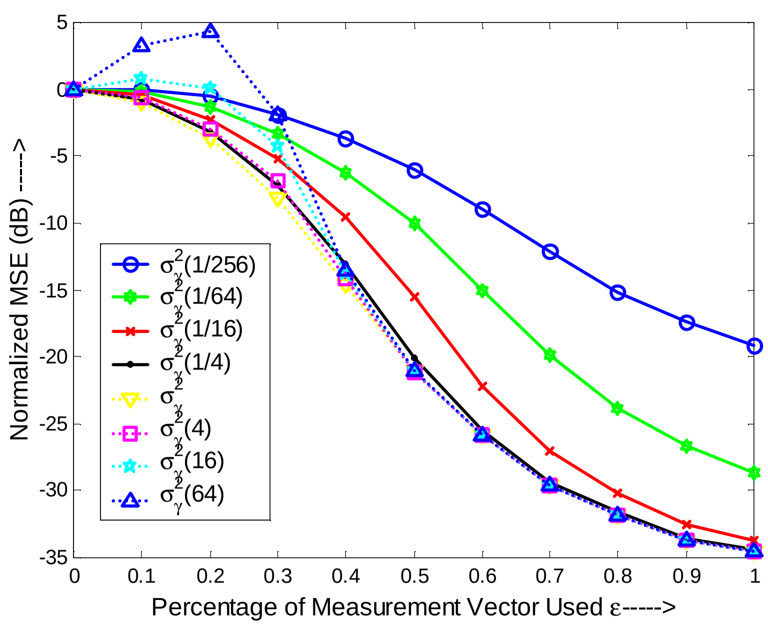
Convergence of the Developed KF for different Initial Conditions

- How does the initial conditions affect the convergence of the Kalman Filter ?

- What if the error covariance matrix is not properly initialized to represent the target and measurement scenario?

- Simulations done for different initial conditions

Convergence vs. Projected Uncertainty



Estimation error vs. the percentage of measurement vector processed

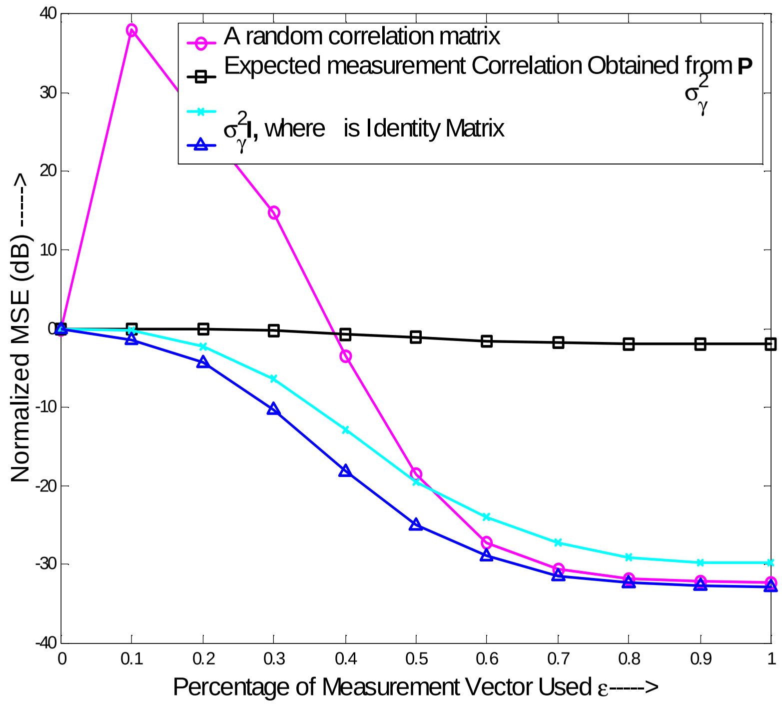
$$\mathbf{K}_\gamma = (c)\sigma_\gamma^2 \mathbf{I}$$

- Error Covariance is initialized as a diagonal Matrix scaled to constant times $|\gamma|^2$
- What if the expected $|\gamma|^2$ is far from the actual value?

Results:

- Convergence is high for values closer to the actual value
- Better to Initialize Error covariance Matrix so as to include large uncertainty region of the estimate

Convergence vs. Estimation Error



Estimation error vs. the percentage of measurement vector processed

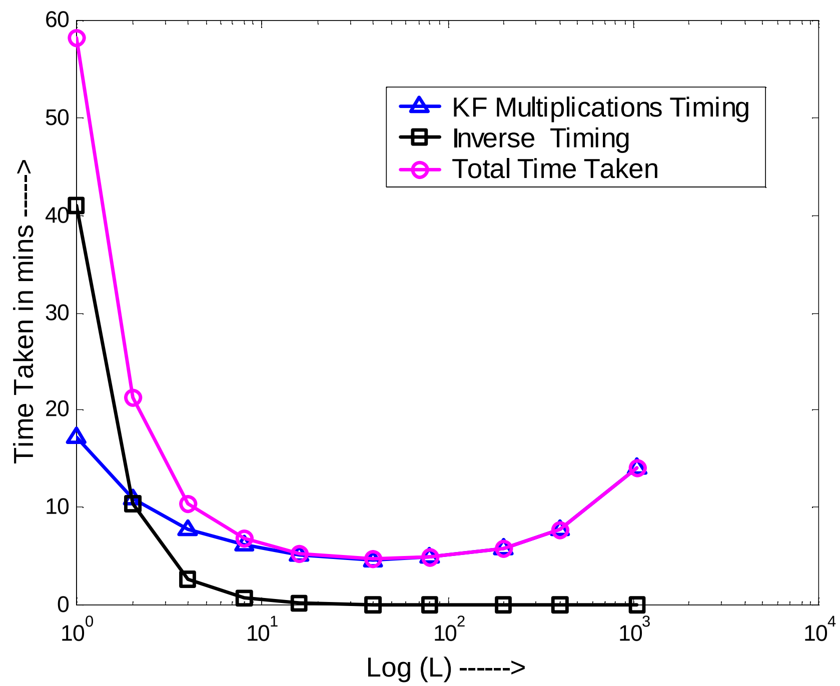
- Able to converge for a random correlation matrix scaled to project a large uncertainty region
- Error covariance matrix is computed based on the Measurement correlation matrix
- Optimal Initialization

Information and Telecommunication Technology Center

Solution to Problem1

Processing Done on bigchips.ittc.ku.edu

CPU 697.7 MHz 12.5 times faster than MMSE



Processing Time vs L

- Iterative implementation using
- Kalman Filter
- Accuracy of estimate maintained
- Initial conditions well defined
- Computational Load Decreased
- Processing speed Increased

Information and Telecommunication Technology Center

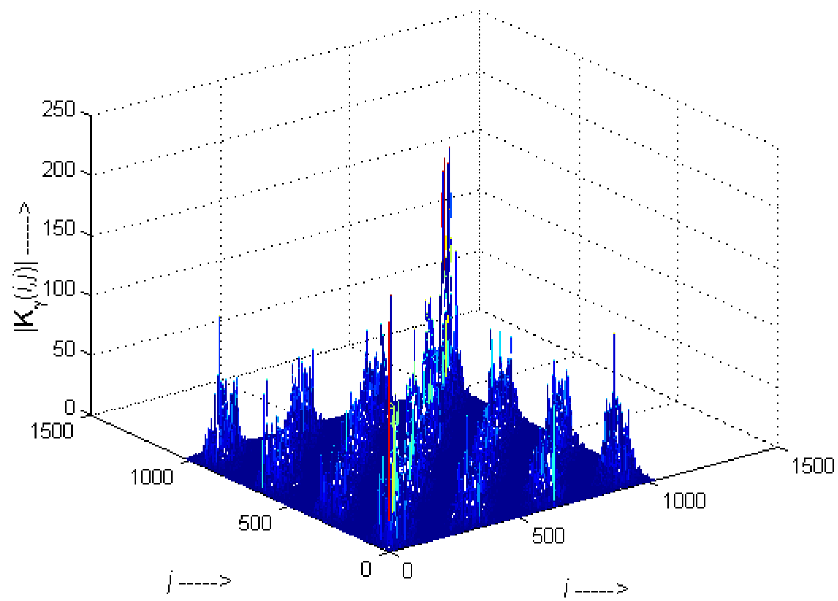
Problem 2: Size of Error Covariance Matrix

- How to reduce the computations defining error covariance matrix ?

- Magnitude of the problem:
  Order of Error Covariance Matrix for a resolution of 256 by 256 is 256^2 by 256^2
  i.e., 4.29 billion points
  Memory Requirement 32GB
  How do you initialize it !!!!???

- Proposed Solution: **Reduce the Rank** of Error Covariance Matrix

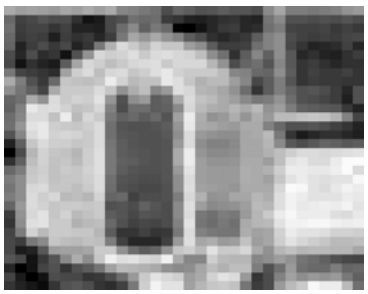
- Is it possible to represent the error correlation in a fewer number of points



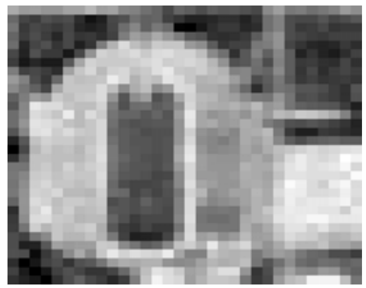
Updated Error Covariance Matrix after using 50% measurement data

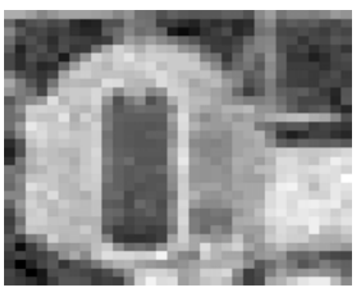
Only Diagonal Elements are considered

Original Image

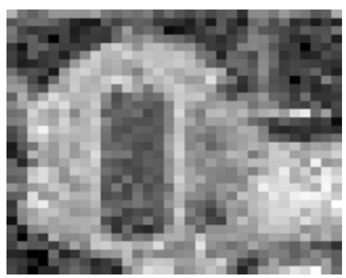


KF Estimate





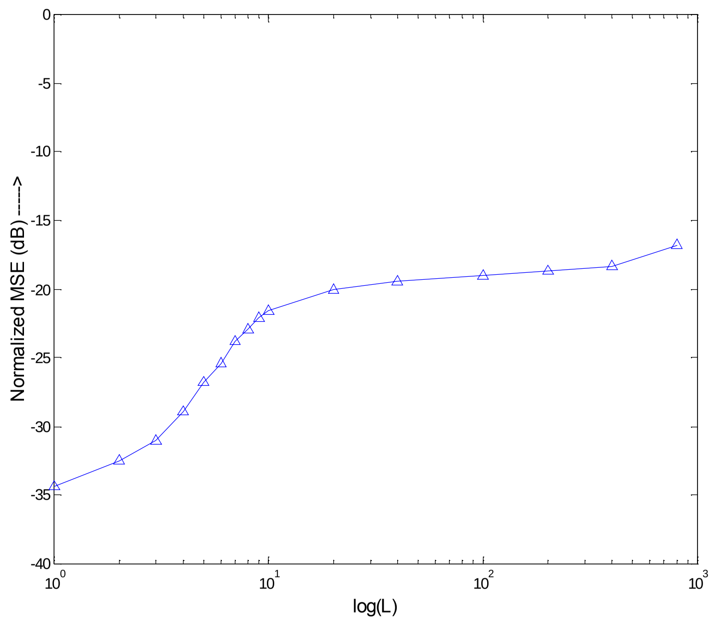
Modified KF
Estimate for L = 4

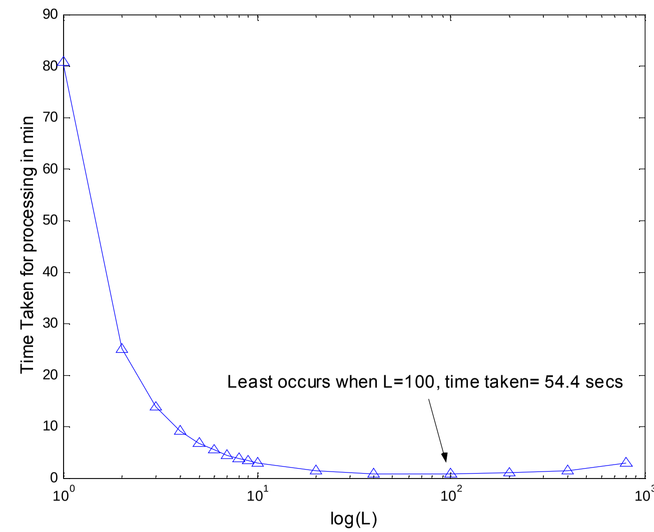


Modified KF
Estimate for L = 100

- Only diagonal points are defined in the Error Covariance Matrix
- Physical representation: Reflectance obtained from a target pixel is uncorrelated to other target pixels.
- It is possible to modify the KF implementation so that only diagonal points are considered in the error covariance matrix

Only Diagonal elements are Considered



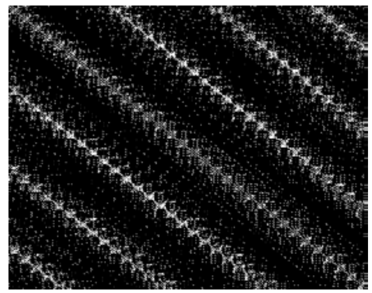


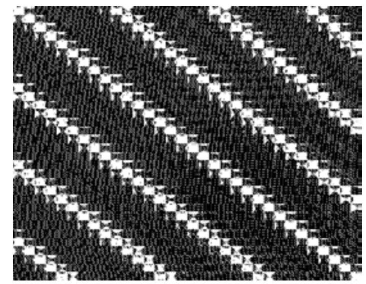
Tradeoff between accuracy and computation load

- Kalman Gain produced is quite close to the one obtained by considering all the elements for smaller L
- During the initial stages of processing diagonal elements are very high final stages noise dominates

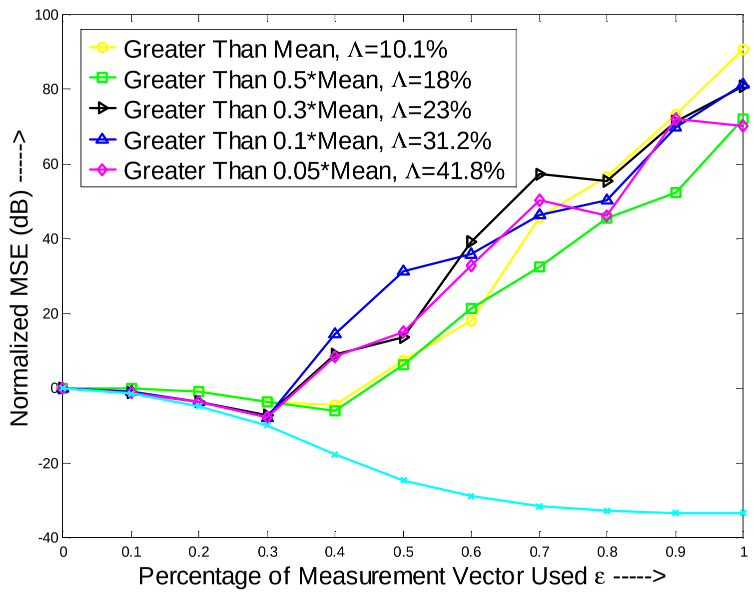
Based on the Measurement Correlation Matrix

Points greater than mean



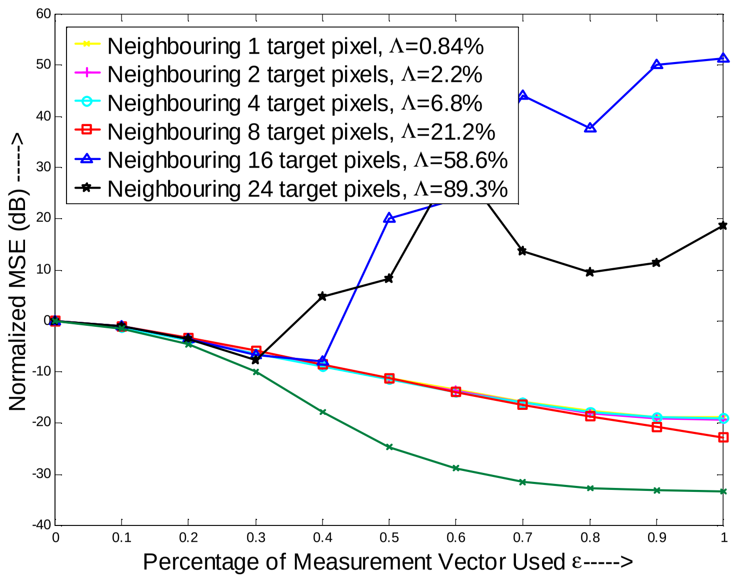


Points greater than 0.1* mean

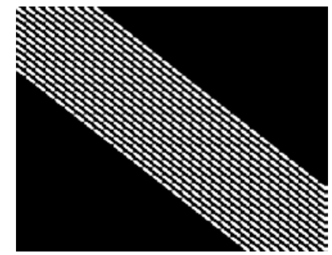


- Can the error covariance be defined by the measurement scenario?
- Points in the measurement correlation matrix signifying high correlation are considered
- Non linear processing errors

Correlation between Targets Closer to each other



General SAR Image



Pts. Considered for 8 neighbors

- Scattering characteristics exhibited by targets closer to each other can be expected to be similar
- Difficult to predict ?
- Useful for mapping regions with target regions exhibiting similar scattering characteristics
- Non Linear Processing Errors Might result

Information and Telecommunication Technology Center

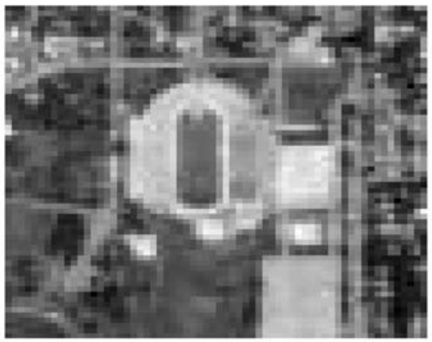
Solution to Problem 2

- Tradeoff offered between accuracy and reduced computational load
- Difficulty level!
- Non linear processing Errors!

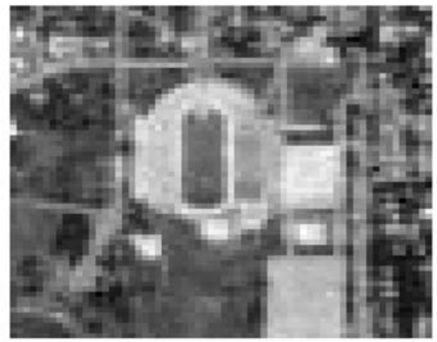
Problem 3: Processing Time

- SAR processing Time
- Proposed Solutions
- Sequential Estimation
- Parallel Processing

Sequential Estimation



Input Image



Final Image Estimate (1)

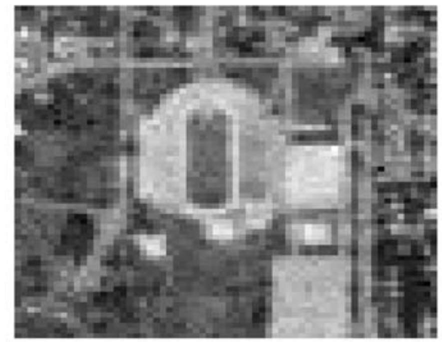
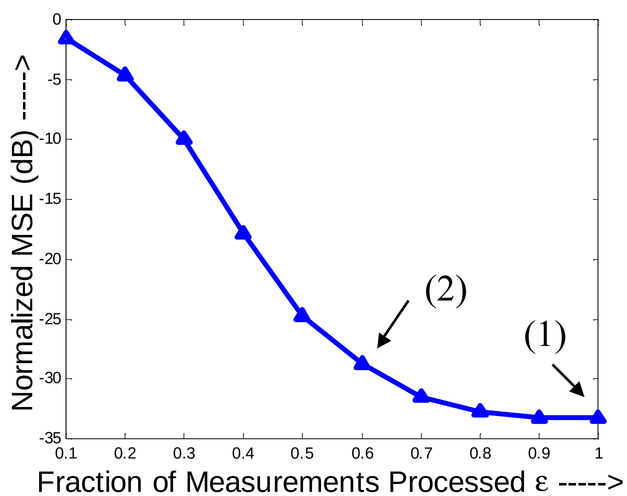


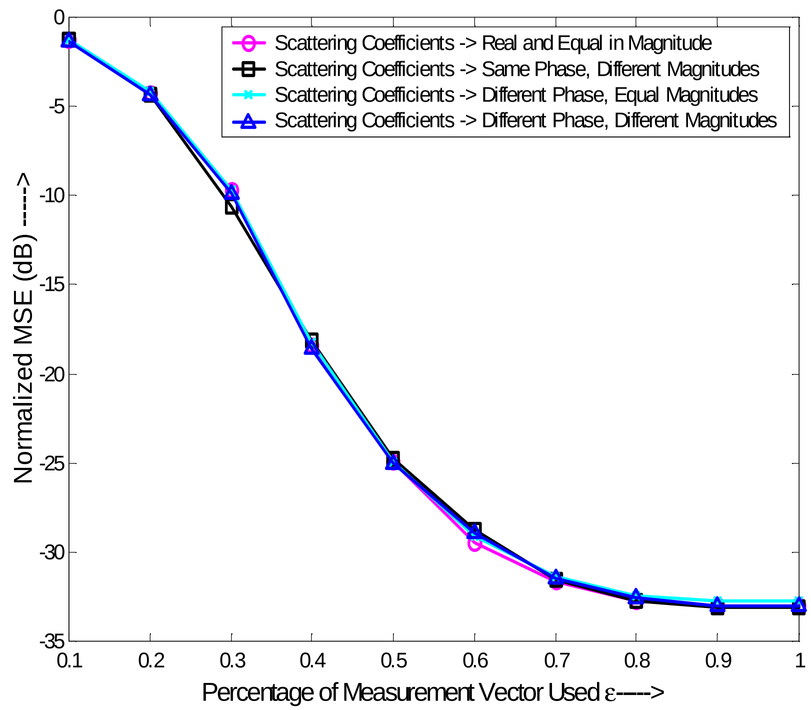
Image Estimate after using 60% Meas. Data (2)



- Improvement in accuracy
- Optimal Estimate
- Sufficient accuracy?
- Processing time saved !

Information and Telecommunication Technology Center

Sequential Estimation

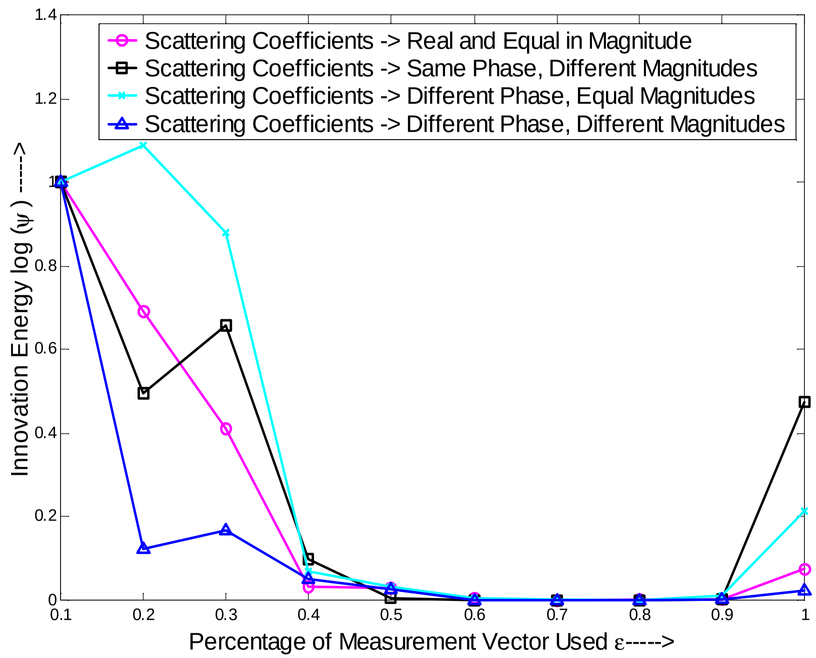


Estimation error vs. the percentage of measurement vector processed for different target scattering scenarios

- How to know when the desired accuracy is reached ?
- Is it possible to quantify the accuracy of the estimate in terms of a parameter in the KF?
- Can this parameter be used for all target scenarios?
- Different Scattering Characteristics
- How does the NMSE vary for the above scattering scenarios

Information and Telecommunication Technology Center

Innovation Energy

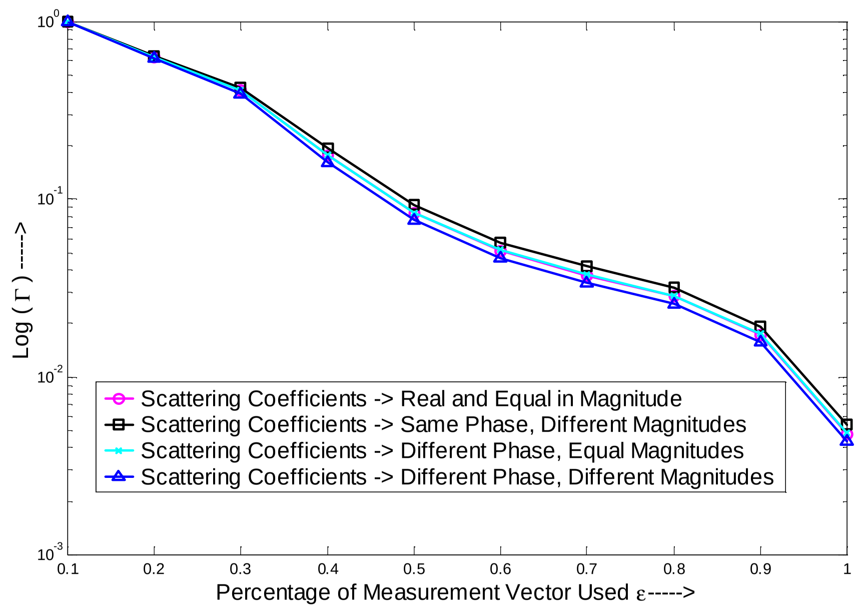


Innovation Energy vs. the percentage
of measurement vector processed
for different target scattering scenarios

- What is Innovation Energy?
- Why should it be considered?
- Results
- Random Component

Information and Telecommunication Technology Center

Kalman Gain

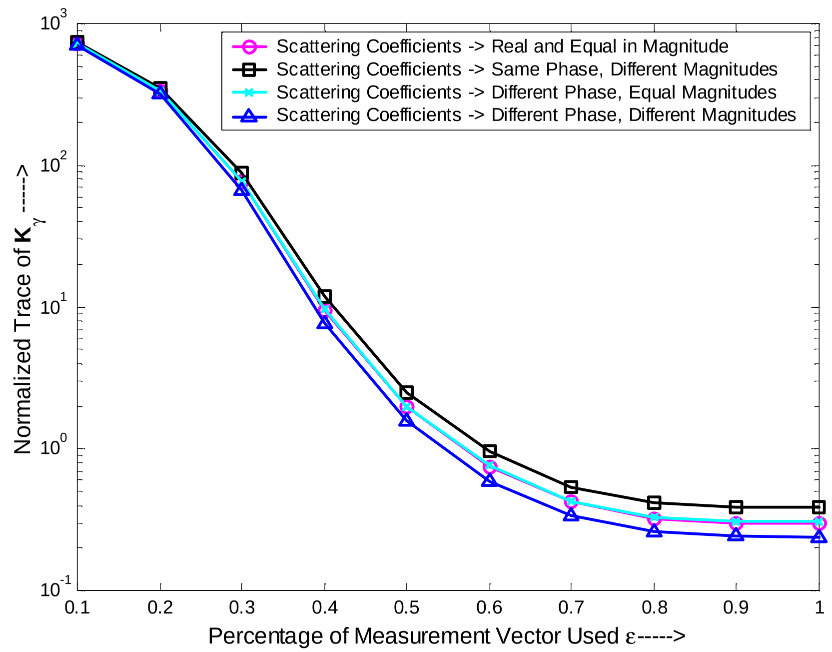


Kalman Gain vs. the percentage
of measurement vector processed
for different target scattering scenarios

- What is Kalman Gain?
- Why should it be considered?
- Results
- Random Component

Information and Telecommunication Technology Center

Trace of the Error Covariance

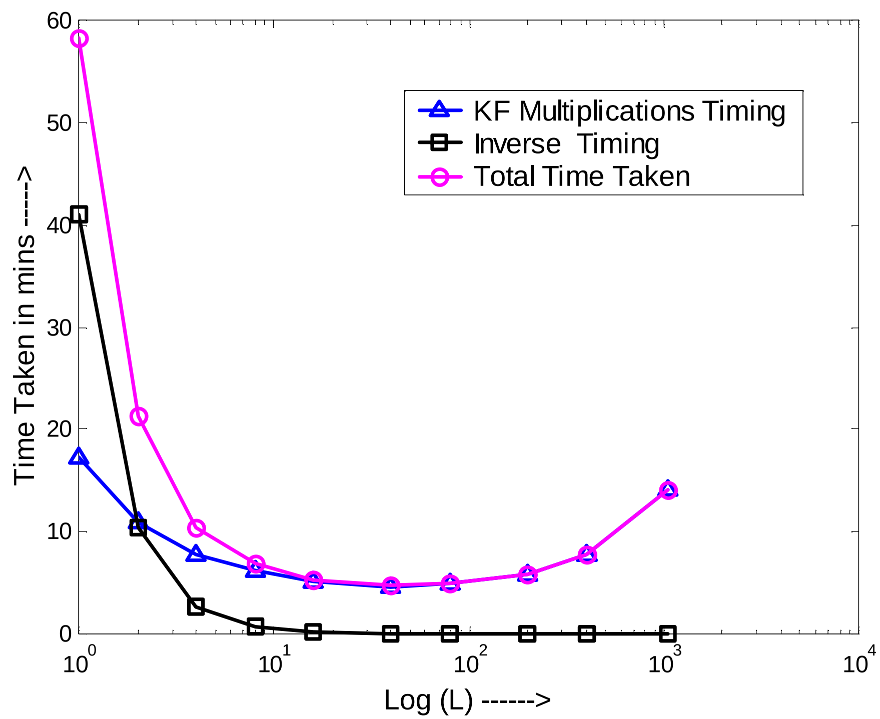


Trace of the Error Covariance vs. the percentage
of measurement vector processed
for different target scattering scenarios

- The sum of the diagonal elements in the Error covariance matrix is computed and normalized to its initial value
- Results
- Variation similar to normalized MSE
- Preset threshold

Information and Telecommunication Technology Center

Matrix Multiplications Time

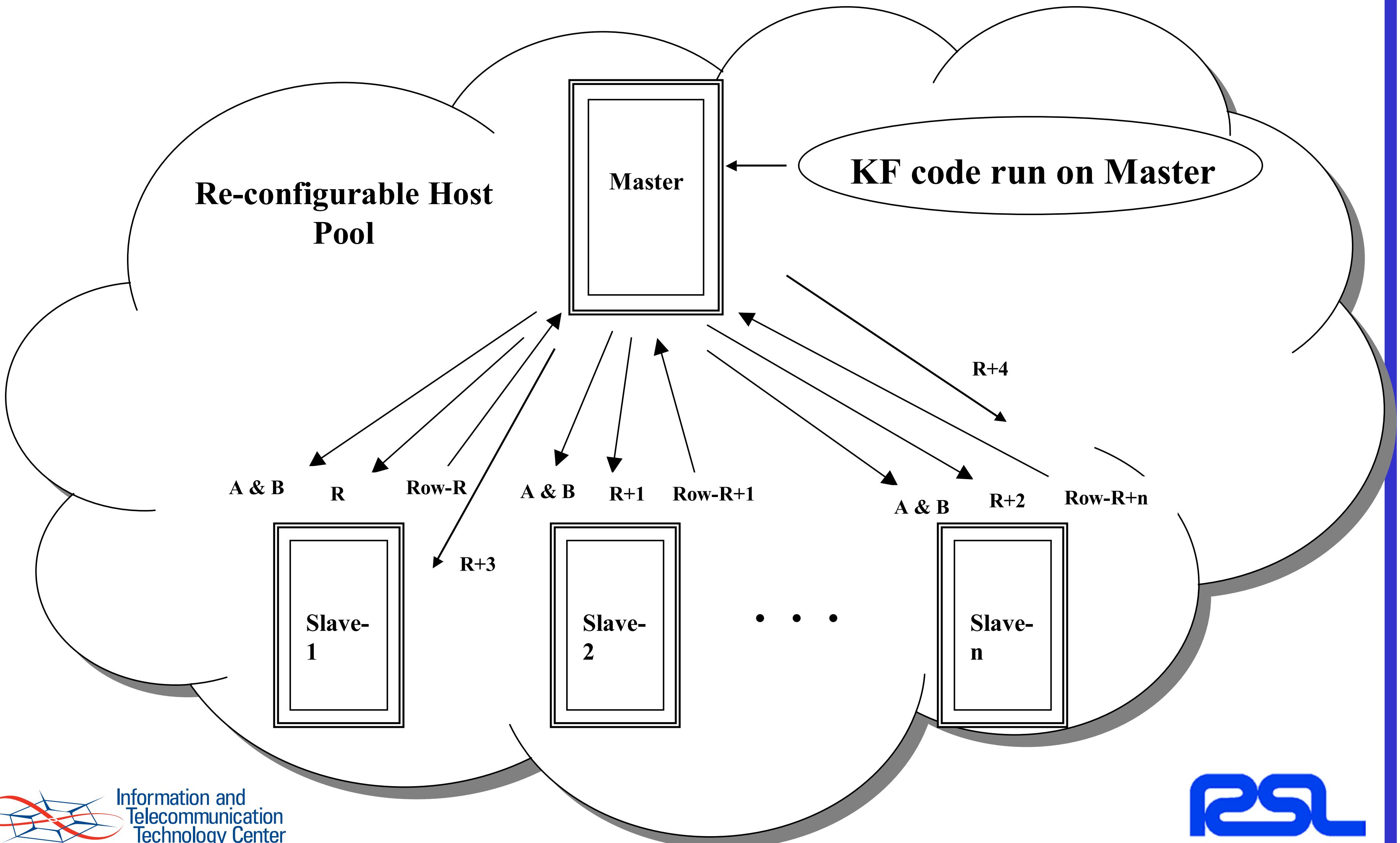


Processing Time vs L

- Initially Inverse Dominates
- Later on Matrix Multiplications Becomes the only factor
- Is there a way to speed up the matrix Multiplications?

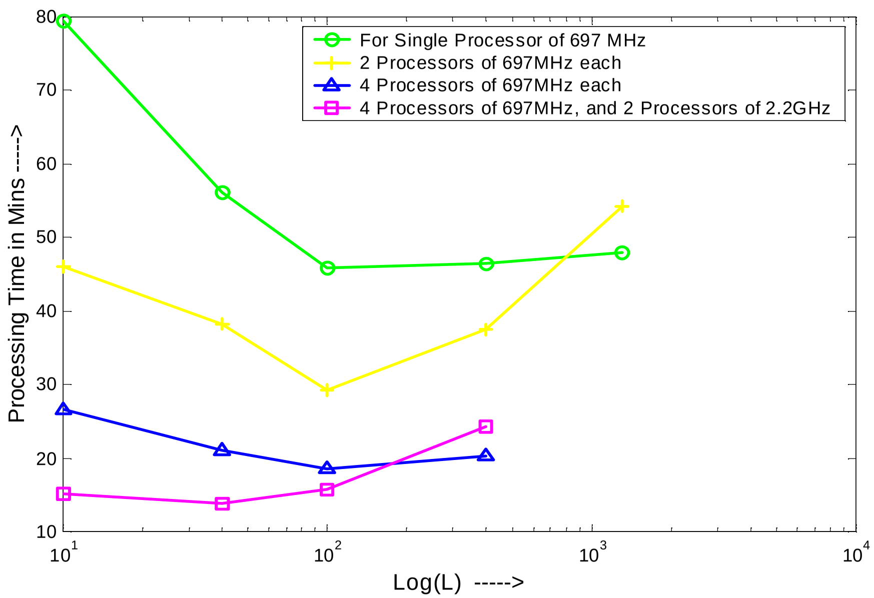
Information and Telecommunication Technology Center

Matrix Multiplication on Parallel Processors



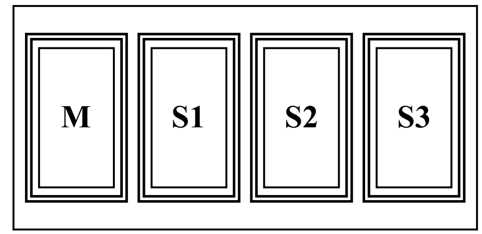
Information and Telecommunication Technology Center

Testing for Parallel Implementation



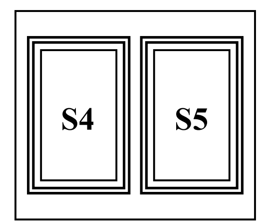
bigchips.ittc.ku.edu

Processors: 697 MHz each



flood.ittc.ku.edu

Processors: 2.2 GHz each



- Resolution of 32 by 32
- Potential of Parallel Processing
- Communication Time

Information and Telecommunication Technology Center

Solution for Problem 3

- Tradeoff offered between accuracy and processing speed
- Matrix multiplications speed increased using parallel processing
- Solution offered for huge matrix sizes

Summary

Problem 1: Huge Computational Load due to Inverse
Solution: Iterative implementation of MMSE using Kalman Filter

Problem 2: Operations involving Huge Error Covariance Matrix
Solution: Tradeoff offered between accuracy and computation load

Problem 3: Huge Processing Time
Solution: Tradeoff Offered between accuracy and processing speed
Parallel Processing

Future Work

- Testing the proposed scenarios on real data
- Incorporating LAPACK (Linear Algebra Package) and BLAS (Basic Linear Algebra Subprograms) routines
- Further testing for different resolution scenarios and different parallel configurations

Questions?