Data Domain STAP with Knowledge-Aided Pre-Whitening

KASSPER Workshop 2004
April 5-7, 2004
Clearwater, FL

Jamie Bergin
Chris Teixeira
Paul Techau

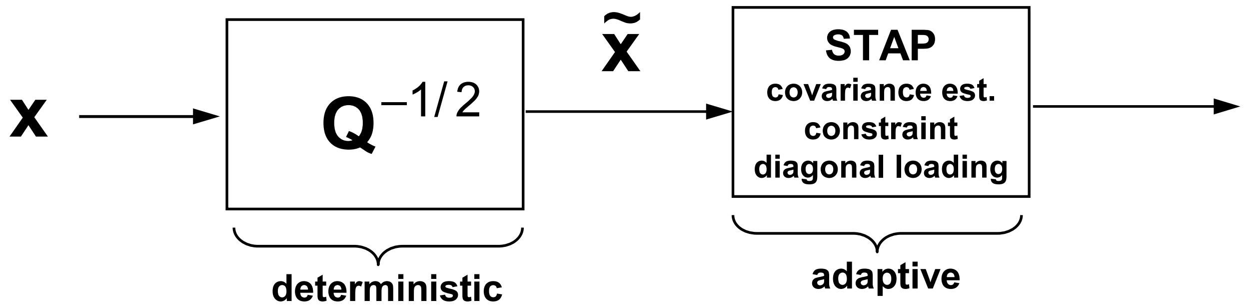
8130 Boone Blvd. Suite 500
Vienna, Virginia 22182
(703)448-1116 FAX: (703)356-3103
www.islinc.com

INFORMATION SYSTEMS LABORATORIES, INC.

Outline

- **Background**
- **Data Domain STAP**
- **Knowledge-aided Pre-Filtering**
  – Algorithm derivation
  – Data domain implementation
  – Computational and communications requirements
- **Summary and Future Work**

Background



- **The desired approach is to cancel the known interference component (data pre-conditioning)**
  - Full ground clutter model
  - Discretes
  - Jammers/RFI
- **Follow up with adaptive processing → will require fewer DoFs**
- **This form does not readily "fit" existing data domain STAP implementations**

Pre-Filter

- **Pre-filter can be something relatively simple like a known discrete (derived from land cover or SAR image)**

$$\mathbf{Q} = |\alpha|^2 \mathbf{v}(\theta, f)\mathbf{v}^H(\theta, f) + \mathbf{I}$$

- **Or something more complex like a full ground clutter covariance model**

$$\mathbf{Q} = \sum_{p=1}^{P_c} |\alpha_p|^2 \mathbf{v}(\theta_p, f_p)\mathbf{v}^H(\theta_p, f_p) \circ \mathbf{T}_p + \mathbf{I}$$

- **Or a combination of both**

Outline

- Background
- **Data Domain STAP**
- Knowledge-aided Pre-Filtering
  - Algorithm derivation
  - Data domain implementation
  - Computational and communications requirements
- Summary and Future Work

Data Domain STAP

- **Problem is to solve linear system of equations to get the weight vector:**

$$\mathbf{w} = \mathbf{R}_s^{-1}\mathbf{s} \leftrightarrow \mathbf{R}_s\mathbf{w} = \mathbf{s}$$

- **Since the sample covariance is typically Hermitian and positive definite one approach is to find its Cholesky decomposition**

$$\mathbf{C}\mathbf{C}^H\mathbf{w} = \mathbf{s}$$

- **The matrix C is lower triangular so the solution can be readily found using a forward and backward substitution**

$$\mathbf{C}\mathbf{y} = \mathbf{s} \rightarrow \mathbf{C}^H\mathbf{w} = \mathbf{y}$$

Data Domain STAP (cont.)

- It turns out that C can be found without forming the sample covariance by performing a QR decomposition (Q unitary, R upper triangular) on the data matrix directly

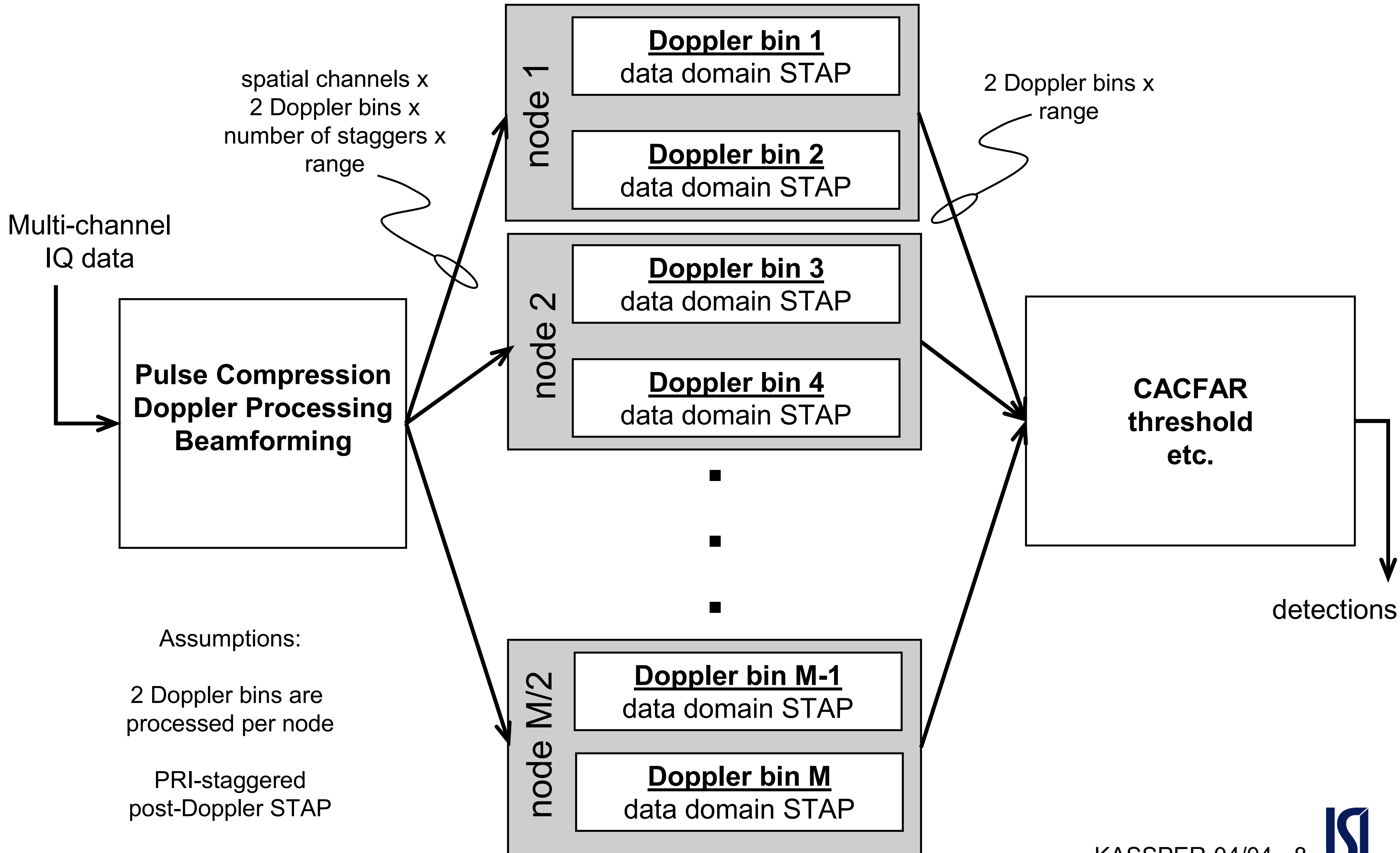
$$\mathbf{X}^H = \mathbf{QR} \Rightarrow \mathbf{R}_s = \mathbf{XX}^H = (\mathbf{QR})^H(\mathbf{QR}) = \mathbf{RR}^H$$

- In this case R is upper triangular and equivalent to the Cholesky decomposition of R_s so it can readily be used to solve for the weight vector (see previous slide)
- It is possible to compute R without explicitly finding Q which saves significant computations
- Diagonal loading can be included by simply augmenting the data matrix

$$\mathbf{X}_a = \begin{bmatrix}\mathbf{X} & \sqrt{\beta}\mathbf{I}\end{bmatrix} \Rightarrow \mathbf{R}_{s,L} = \mathbf{X}_a\mathbf{X}_a^H = \mathbf{R}_s + \beta\mathbf{I}$$

- In general any solution that adds a positive semi-definite matrix to the covariance can be implemented by an appropriate augmentation of the data matrix in the data domain implementation

STAP Algorithm Partitioning Example



Assumptions:

2 Doppler bins are processed per node

PRI-staggered post-Doppler STAP

Outline

- Background
- Data Domain STAP
- **Knowledge-aided Pre-Filtering**
  - Algorithm derivation
  - Data domain implementation
  - Computational and communications requirements
- Summary and Future Work

Reduced-DoF STAP

- **For many systems it is not possible to use full-DoF STAP due to limited computational resources and sample support**
- **A common approach is to break the full-DoF problem into a number of smaller reduced-DoF problems via an** *NM*x*D* (*D* < *NM*) **transformation $\mathbf{H}_m$ on the data:**

$$\mathbf{x}_m = \mathbf{H}_m^H \mathbf{x} \qquad \mathbf{w}_m = \kappa \mathbf{R}_m^{-1} \mathbf{v}_m$$

$$\mathbf{v}_m = \mathbf{H}_m^H \mathbf{v} \qquad y_m = \mathbf{w}_m^H \mathbf{x}_m$$

- **The transformation $\mathbf{H}_m$ can also be applied to the clutter covariance model and thermal noise:**

$$\mathbf{R}_{c,m} = \mathbf{H}_m^H \mathbf{R}_c \mathbf{H}_m \qquad \mathbf{R}_{n,m} = \mathbf{H}_m^H \mathbf{R}_n \mathbf{H}_m$$

thermal noise covariance

- **As an example, multi-bin element space post-Doppler STAP**

$$\mathbf{R}_n = \mathbf{I}_{NMxNM} \qquad \mathbf{R}_{n,m} = \mathbf{H}_m^H \mathbf{H}_m = \mathbf{I}_{DxD}$$

Knowledge-Aided Quadratic Constraints

Reduced-DoF STAP

- We can incorporate the reduced-DoF covariance model as a quadratic constraint

$$\min_{\mathbf{w}_m} \; E\{|\mathbf{w}_m^H \mathbf{x}_m|^2\} \quad s.t. \quad \begin{cases} \mathbf{w}_m^H \mathbf{v}_m = 1 \\ \mathbf{w}_m^H \mathbf{R}_{c,m} \mathbf{w}_m \le \delta_{d,m} \\ \mathbf{w}_m^H \mathbf{w}_m \le \delta_{L,m} \end{cases}$$

want weights to be "orthogonal" to the reduced-DoF *a priori* clutter model

this is the KA part

- Gives:

$$\mathbf{w}_m = \frac{(\mathbf{R}_m + \beta_{d,m}\mathbf{R}_{c,m} + \beta_{L,m}\mathbf{I})^{-1}\mathbf{v}_m}{\mathbf{v}_m^H(\mathbf{R}_m + \beta_{d,m}\mathbf{R}_{c,m} + \beta_{L,m}\mathbf{I})^{-1}\mathbf{v}_m} = \frac{(\mathbf{R}_m + \mathbf{Q}_m)^{-1}\mathbf{v}_m}{\mathbf{v}_m^H(\mathbf{R}_m + \mathbf{Q}_m)^{-1}\mathbf{v}_m}$$

"colored loading"

- Same form as full-DoF case
- Can also be shown to be a prefilter on the reduced DoF data
- Can be implemented in the data domain

Pre-Filter Interpretation

- Colored loading beamformer can be expressed as:

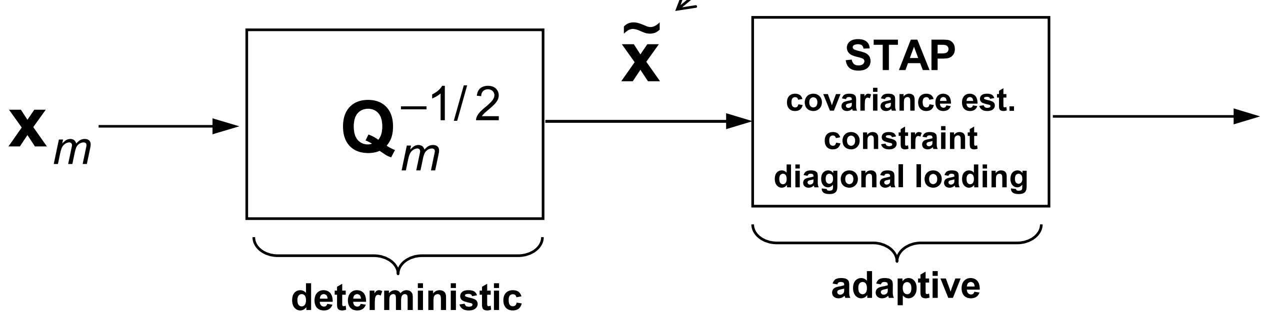
$$\mathbf{w} = \frac{\mathbf{Q}_m^{-1/2}(\mathbf{Q}_m^{-1/2}\mathbf{R}_m\mathbf{Q}_m^{-1/2} + \mathbf{I})^{-1}\mathbf{Q}_m^{-1/2}\mathbf{v}_m}{\mathbf{v}_m^H\mathbf{Q}_m^{-1/2}(\mathbf{Q}_m^{-1/2}\mathbf{R}_m\mathbf{Q}_m^{-1/2} + \mathbf{I})^{-1}\mathbf{Q}_m^{-1/2}\mathbf{v}_m}$$

- This filtering solution is equivalent to *deterministic* pre-filtering followed by *adaptive* processing (i.e., 2 stages)

$$\tilde{\mathbf{x}} = \mathbf{Q}_m^{-1/2}\mathbf{x}_m \quad \tilde{\mathbf{v}} = \mathbf{Q}_m^{1/2}\mathbf{v}_m$$

$$\tilde{\mathbf{w}} = \frac{(\mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} + \mathbf{I})^{-1}\tilde{\mathbf{v}}}{\tilde{\mathbf{v}}^H(\mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} + \mathbf{I})^{-1}\tilde{\mathbf{v}}}$$

it will generally be easier to estimate the interference covariance of the pre-filtered data than the original data because it is likely to have a lower *effective* rank



Data Domain Implementation

- **Typically, the loading matrix $\mathbf{Q}_m$ is Hermitian and positive-definite so that its Cholesky decomposition exists**

$$\mathbf{Q}_m = \beta_{\mathrm{d}}\mathbf{R}_{c,m} + \beta_{\mathrm{L}}\mathbf{I} = \mathbf{Q}_m^{1/2}\mathbf{Q}_m^{1/2} = \mathbf{C}_m\mathbf{C}_m^H$$

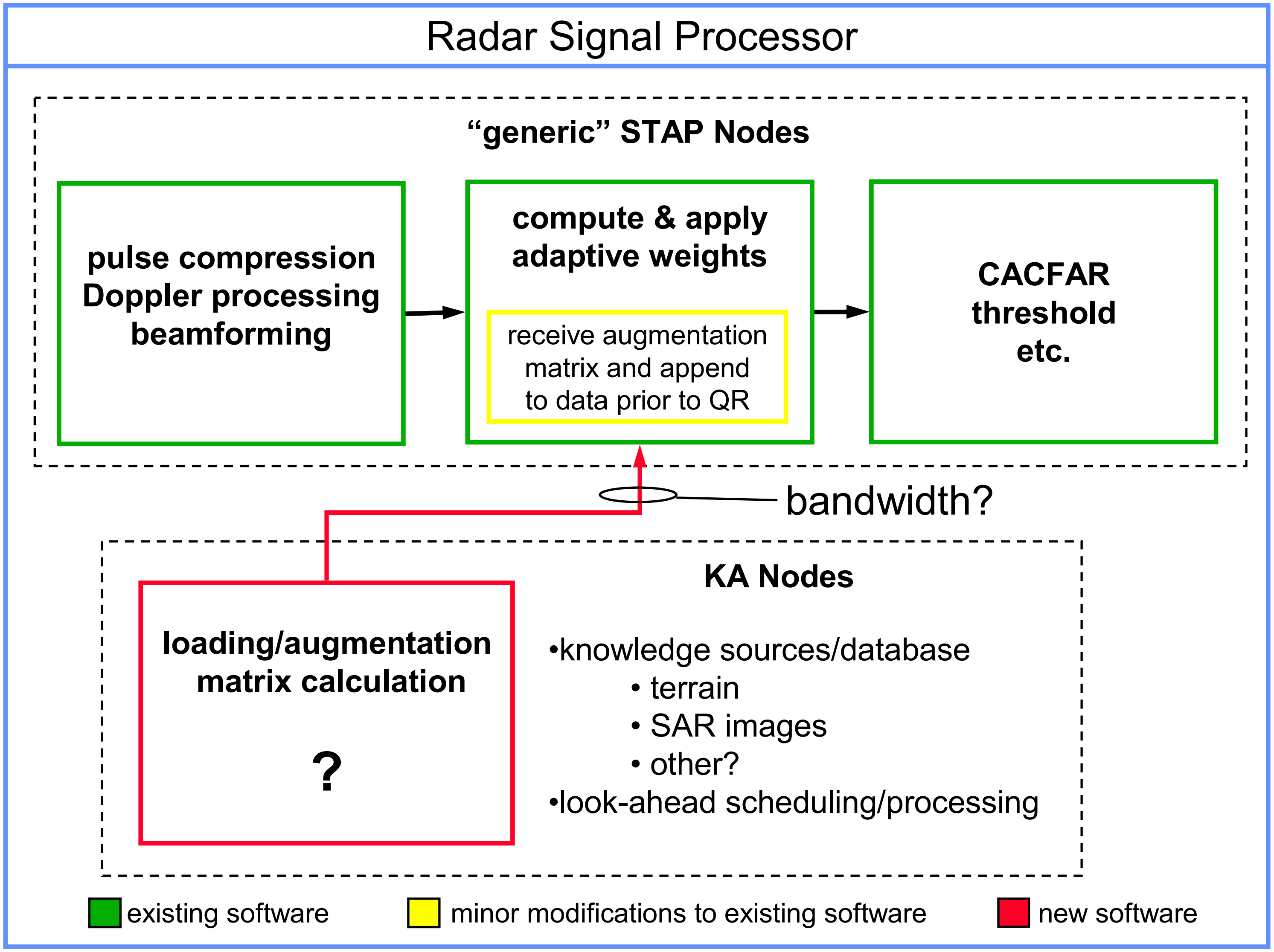
$\mathbf{C}_m$ is lower triangular

- **Can be efficiently implemented in the data domain by augmenting data matrix and using QR decomposition**

$$\mathbf{X}_a = [\mathbf{X}_m \quad \sqrt{K}\mathbf{C}] \rightarrow \frac{1}{K}\mathbf{X}_a\mathbf{X}_a^H = \mathbf{R}_s + \mathbf{Q}_m$$

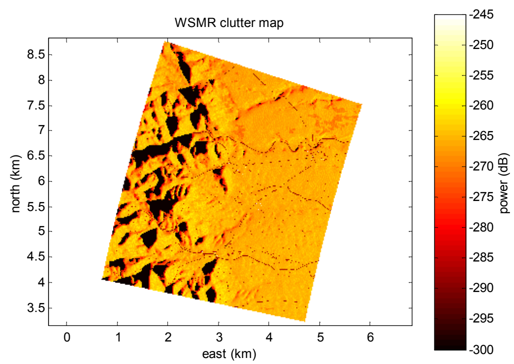
- **Thus, pre-filter approach implemented as colored loading actually "fits" existing STAP computing architectures (if they already employ data domain processing with diagonal loading)**

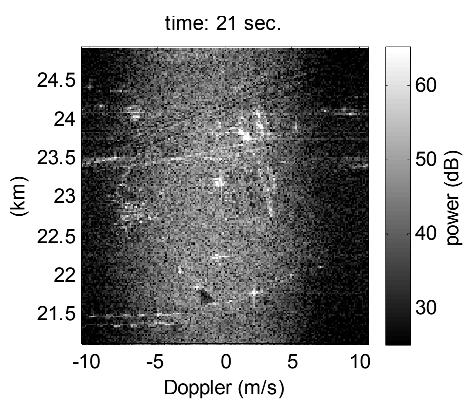
Architecture



Loading Matrix

$$\mathbf{Q}_m = \sum_{p=1}^{P_c} \left|\alpha_p\right|^2 \mathbf{v}_m(\theta_p, f_p)\mathbf{v}_m^H(\theta_p, f_p) + \mathbf{I}$$





site-specific clutter model

own-ship low-resolution SAR

prior CPI clutter maps

$\alpha_p = 1$

deterministic space-time model of the array

calibration table

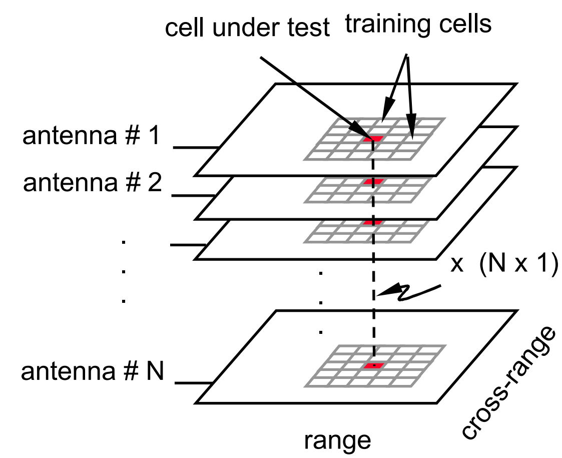
calibration on clutter

multi-channel own-ship low-resolution SAR

$$\mathbf{v}_m(\theta_p, f_p) = (\mathbf{H}_m^H \mathbf{t}(f_p)) \otimes \mathbf{s}(\theta_p)$$

$$\mathbf{t}_{[m]}(f_p) = \exp(j2\pi m f_p T_{pri})$$

$$\mathbf{s}(\theta_p) = ?$$



Loading Matrix

- **For now assume that the elements required to compute the loading matrix are known (steering vectors and power coefficients)**
- **Computations to compute loading matrix and perform Cholesky decomposition:**

$$FLOPS = 8P_c D^2 + \frac{4}{3} D^3$$

D - # of DoFs

P_c - # of patches

- **Computations to compute QR decomposition of data matrix augmented with Cholesky factor of loading matrix:**

$$FLOPS = 8D^2 K$$

K - # of data samples

- **Cost of computing loading matrix and its Cholesky decomposition is expensive → especially if it changes vs. range**

Loading Matrix

- **Since we want the augmentation to result in both colored and diagonal loading components**

$$\mathbf{R} = \mathbf{R}_s + \beta_{\mathrm{d}}\mathbf{R}_c + \beta_{\mathrm{L}}\mathbf{I}$$

- **Augmenting the data matrix with the scaled steering vectors and identity matrix directly will give the desired result**

$$\mathbf{X}_a = \left[\mathbf{X}_m \quad \sqrt{\beta_{d,m}}\mathbf{V}_m \quad \sqrt{\beta_L}\mathbf{I}\right] \quad p^{th} \text{ column of } \mathbf{V}_m = \left|\alpha_p\right|\mathbf{v}_m(\theta_p, f_p)$$

- **Computations to compute QR decomposition of this augmented data matrix:**

$$FLOPS = 8D^2(K + P_c)$$

- **Results in a savings of ~**$(4/3)D^3$ **computations per range and Doppler bin**
- **Moves major computational burden to the STAP nodes**

Loading Matrix (updating)

- In practice it has been found that a single colored loading matrix for underlying distributed ground clutter will work over large range swaths for stand-off geometries
- Thus, the augmentation matrix will occur once for ground clutter
- However, it will likely be desirable to include contributions from discretes (e.g., derived from SAR images) at various range bins
- This can be accommodated by simply augmenting the new data matrix with an extra column

$$\mathbf{X}_a = \left[\mathbf{X}_m \quad \sqrt{\beta_{d,m}}\mathbf{V}_m \quad \sqrt{\beta_L}\mathbf{I} \quad \sqrt{\beta}\mathbf{d}\right]$$

steering vector
of a discrete

- Adding (or subtracting) a column and updating the QR decomposition can be done efficiently
- In fact an updating procedure will probably be available to accommodate sliding window/hole training strategies

Bandwidth Requirements

- **Assumptions:**
  - One complex DxP_c loading/augmentation matrix is required per Doppler bin per CPI
  - 128 pulses with 2x oversampling → 256 Doppler bins
  - 100 ms CPI length
  - 6 DoFs and 20 patches
  - Double precision real and imaginary data
- **Bandwidth required to communicate augmentation matrices**

$$BW = \frac{(64\text{bits})(256)DP_c}{100\text{ms}} = 9.8\text{Mbps}$$

- **May want to only apply this technique to a small number of Doppler bins**
- **Move some of the calculations onto the STAP nodes?**

Outline

- Background
- Data Domain STAP
- Knowledge-aided Pre-Filtering
  - Algorithm derivation
  - Data domain implementation
  - Computational and communications requirements
- **Summary and Future Work**

Summary and Future Work

- Data domain STAP was reviewed → based on QR decomposition of the data matrix
- Adding matrix to covariance can be implemented in the data domain by augmenting the data matrix
- Pre-filtering is equivalent to colored loading → offers an efficient way to implement in existing data domain STAP beamformers
- Alternatives for computing the augmentation matrix were considered → direct augmentation with covariance model steering vectors appears to be most efficient
- Bandwidth requirements appear reasonable
- Next step is to begin coding the technique for integration on the KASSPER test bed computer
- Continue to assess performance gains of knowledge-aided beamforming → KASSPER data sets and Tuxedo data