KASSPER Conference

Signal Processing Algorithms for KASSPER

Charlie Rader

Nick Pulsone

Dan Kreithen

April 3, 2002

50 years

MIT LINCOLN LAB

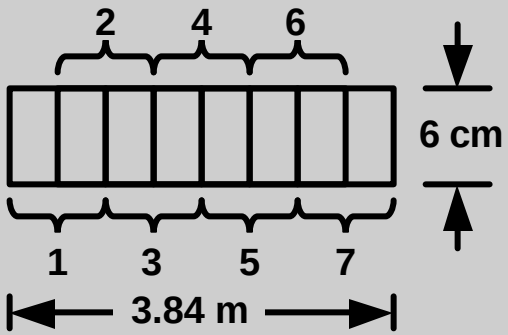
Outline

- Introduction
- Baseline STAP Algorithm Description
- Adequate Gain Constraints
- KASSPER Algorithms
- Summary

GMTI Radar System Assumptions

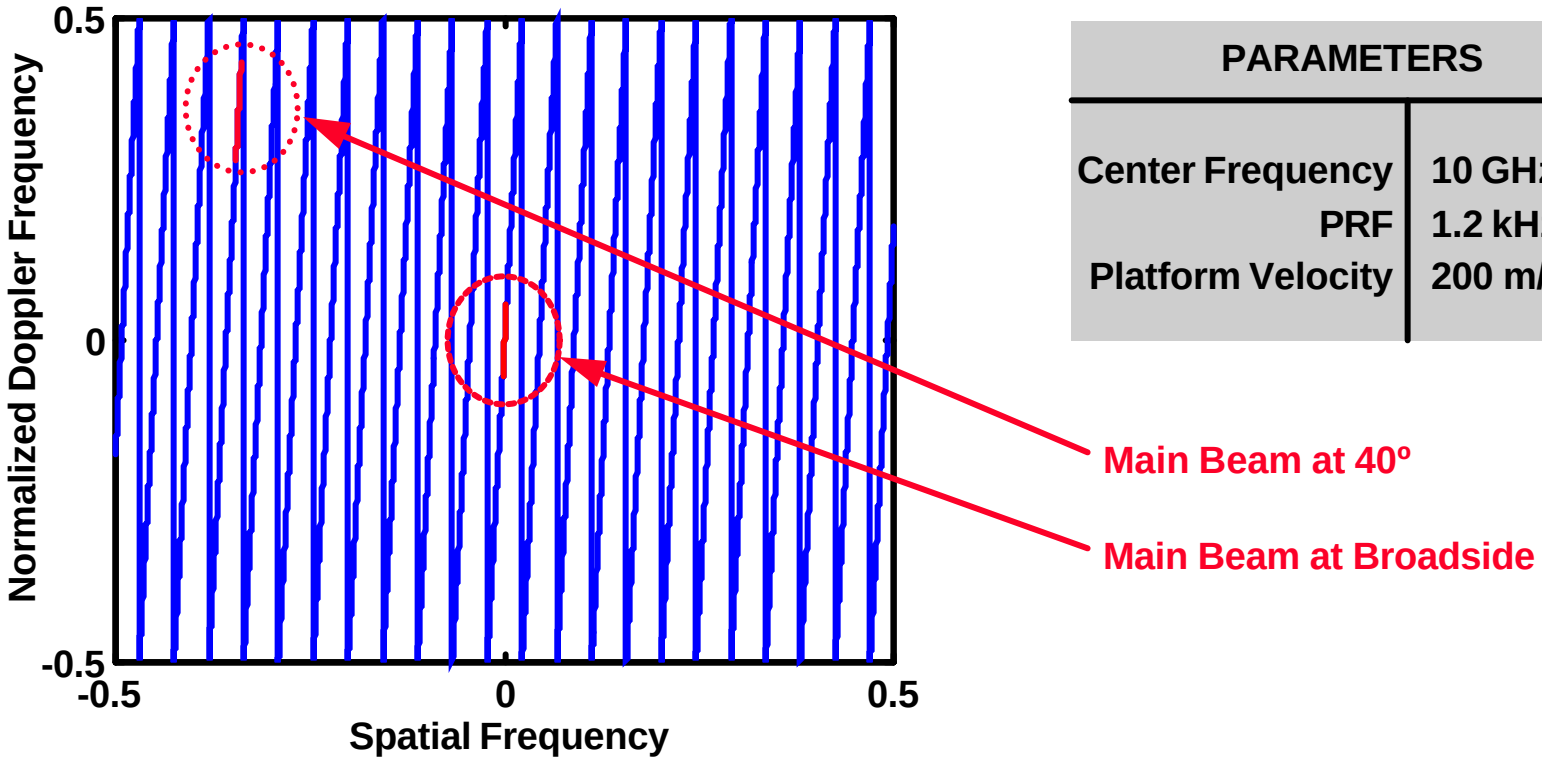
PARAMETERS	
Center Freq.	10 GHz
PRF	1.2 kHz
Sample Rate	20 MHz
Range	112 km
Pulses	32
Duty Cycle	10%

ANTENNA ARRAY



- 9 Channels
  - 7 overlapping subarrays
  - 2 auxiliaries
- Transmit Pattern?
- Transmit Power?

Clutter Spectral Response



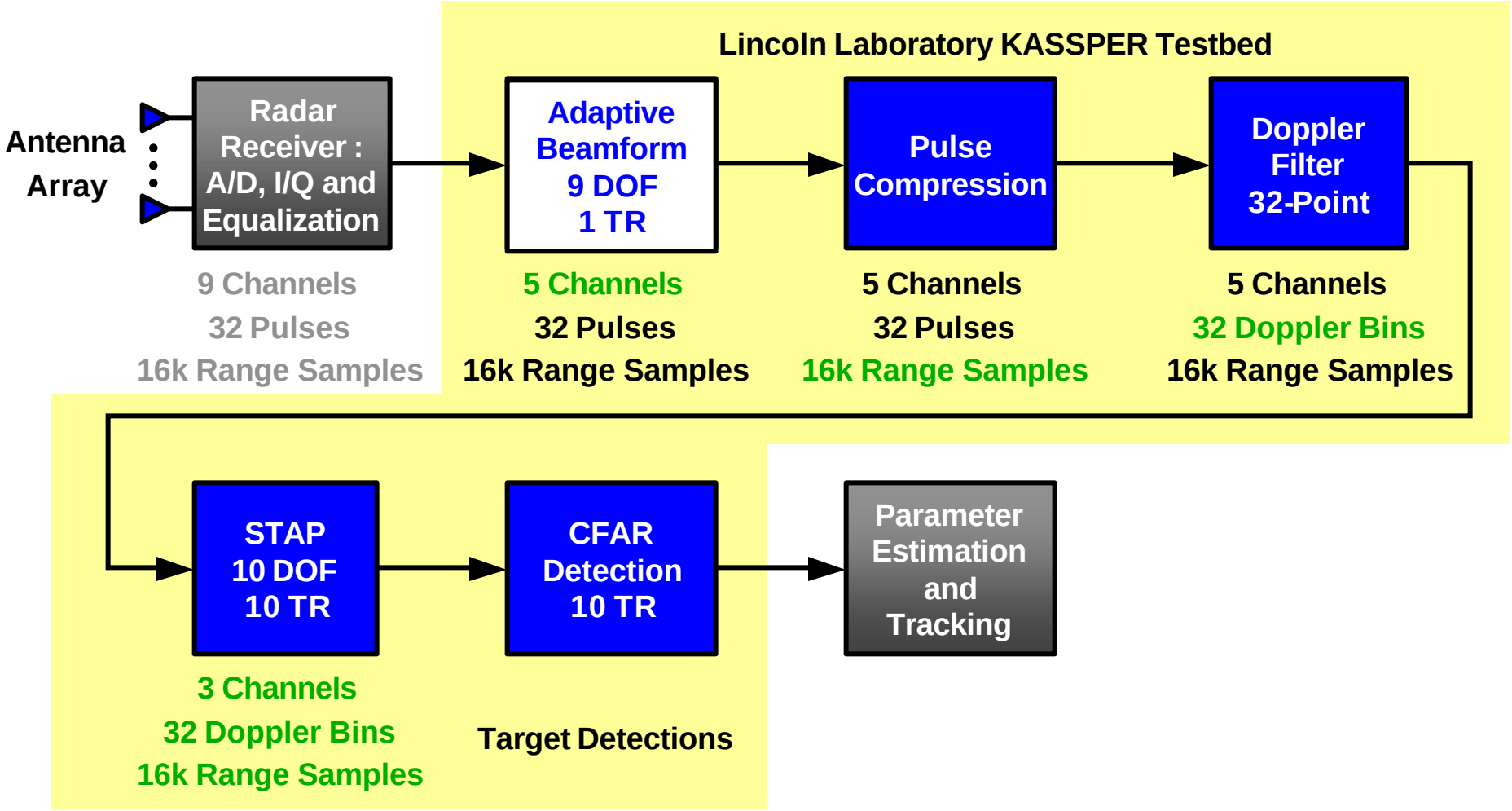
PARAMETERS	
Center Frequency	10 GHz
PRF	1.2 kHz
Platform Velocity	200 m/sec

Max. Unambiguous Velocity	± 9 m/sec
Doppler Ambiguity ?	22.2

Outline

- Introduction
- Baseline STAP Algorithm Description
- Adequate Gain Constraints
- KASSPER Algorithms
- Summary

Baseline STAP Architecture



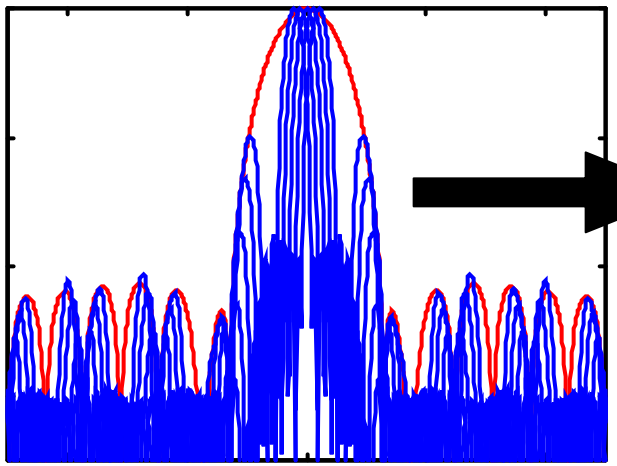
Estimated Sustained Throughput 33.7 Gflop/s

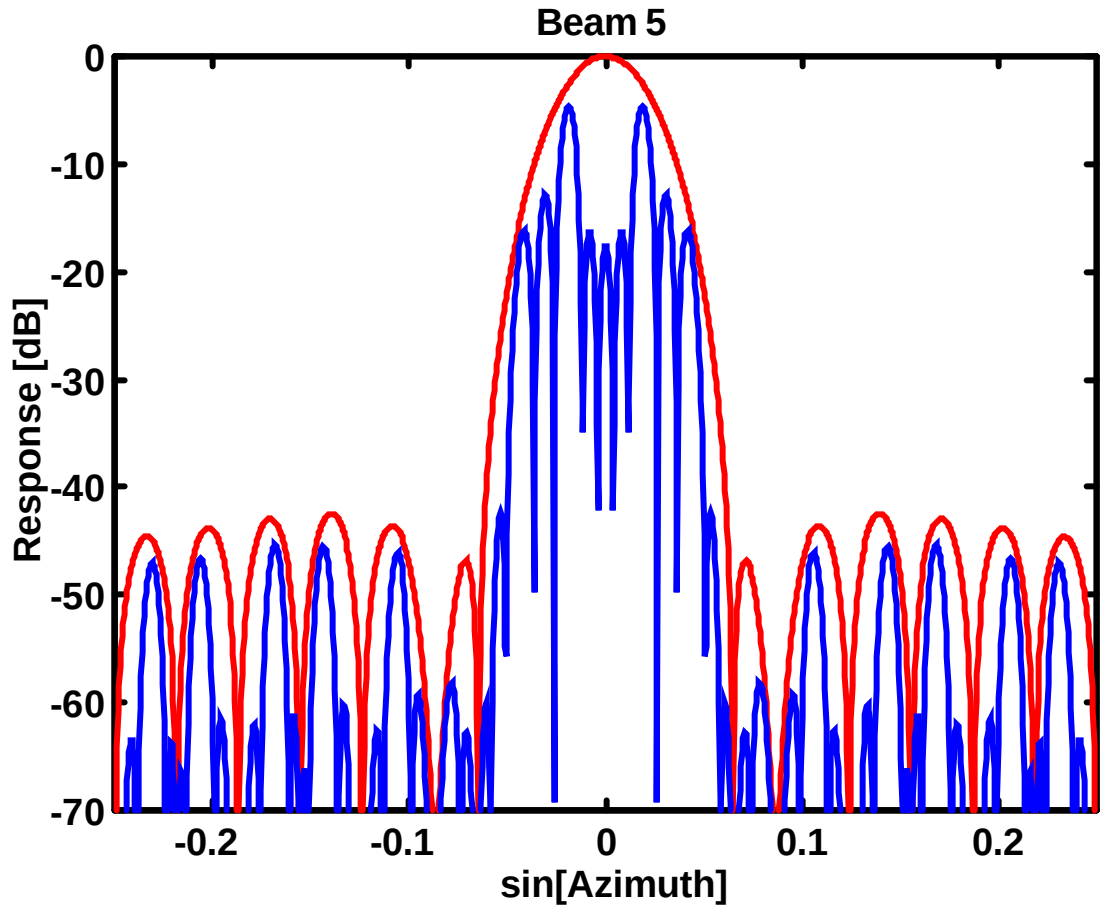
Estimated Peak Throughput Required 340 Gflop/s

Ideal Eigen-Beam Response

(a.k.a Prolate Beams)

Define Signal Subspace



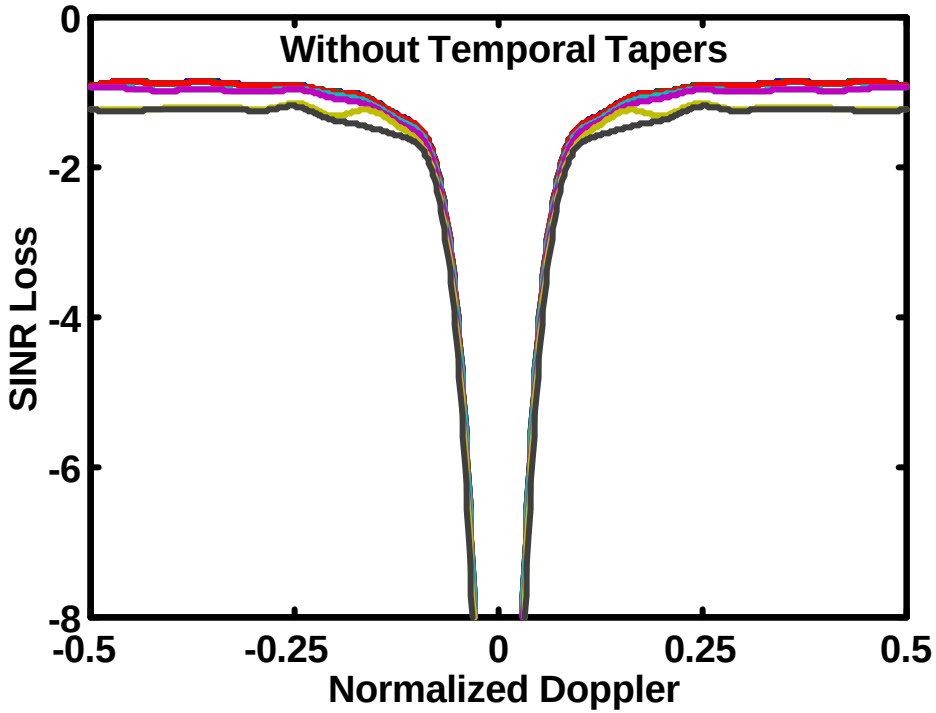


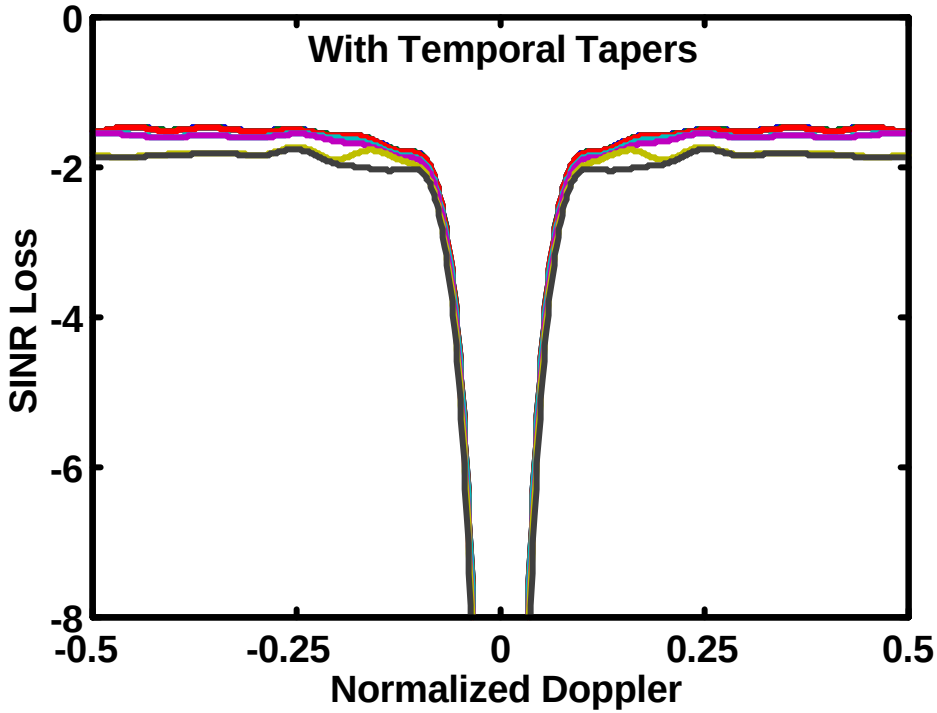
Subarray 3dB-Beamwidth 2.35°

Full Array 3dB-Beamwidth 0.92°

Assumes Hamming Weighting

SINR Loss Performance for Different Beamspace Dimensions





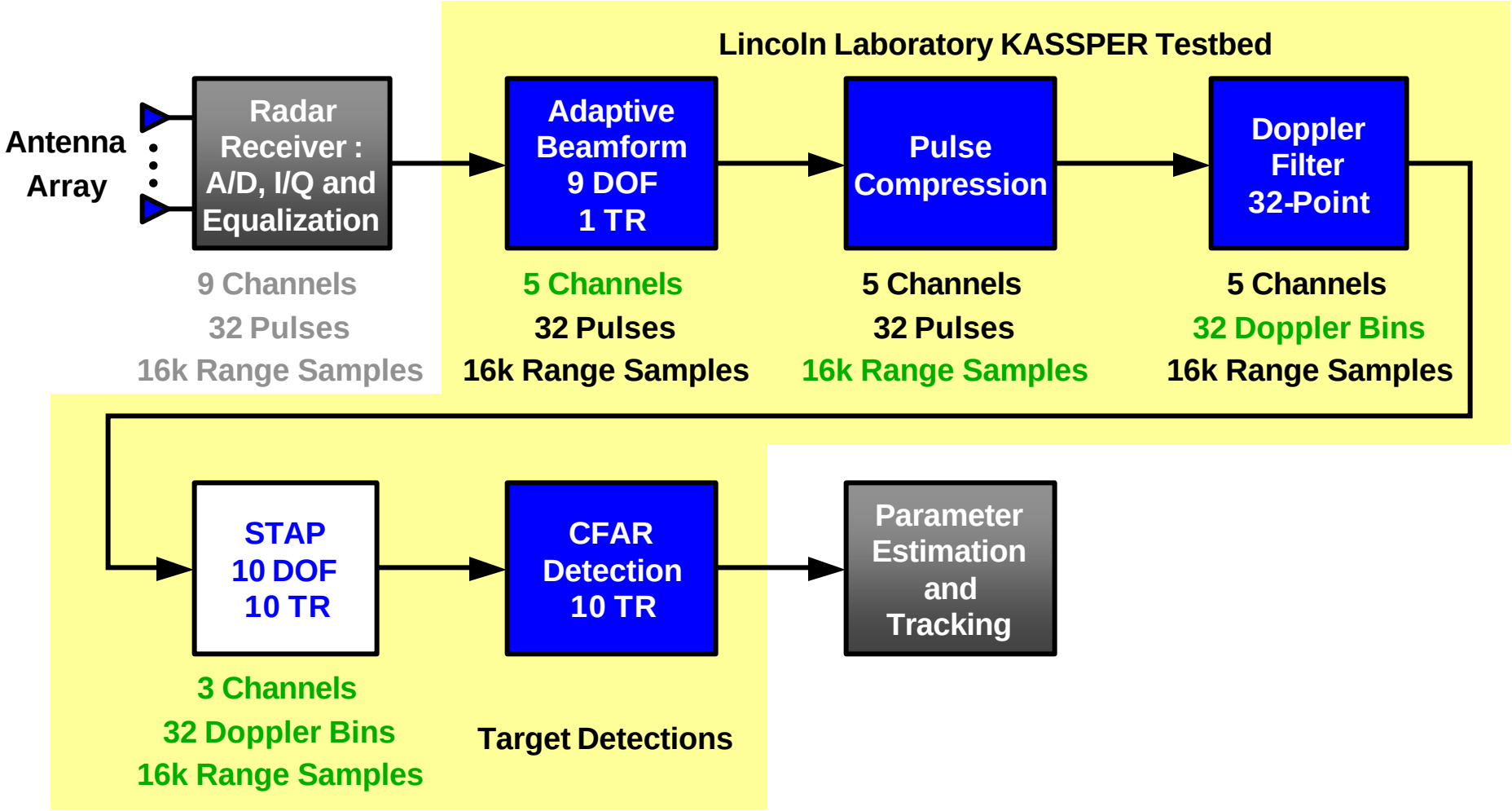
- Fully Adaptive, UFDS = 0.915
- Beamspace K=9, UFDS = 0.915
- Beamspace K=8, UFDS = 0.915
- Beamspace K=7, UFDS = 0.915
- Beamspace K=6, UFDS = 0.912
- Beamspace K=5, UFDS = 0.908
- Beamspace K=4, UFDS = 0.908
- Beamspace K=3, UFDS = 0.905

CNR = 60dB
Azimuth = 0º
Uniform Clutter Distribution

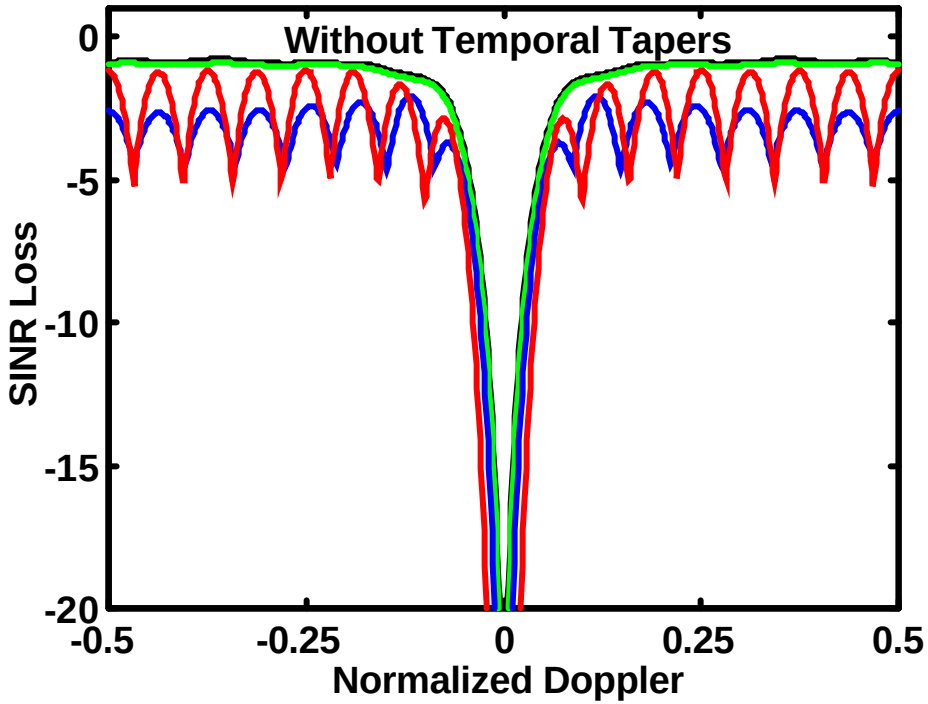
- Fully Adaptive, UFDS = 0.905
- Beamspace K=9, UFDS = 0.905
- Beamspace K=8, UFDS = 0.905
- Beamspace K=7, UFDS = 0.905
- Beamspace K=6, UFDS = 0.905
- Beamspace K=5, UFDS = 0.902
- Beamspace K=4, UFDS = 0.902
- Beamspace K=3, UFDS = 0.899

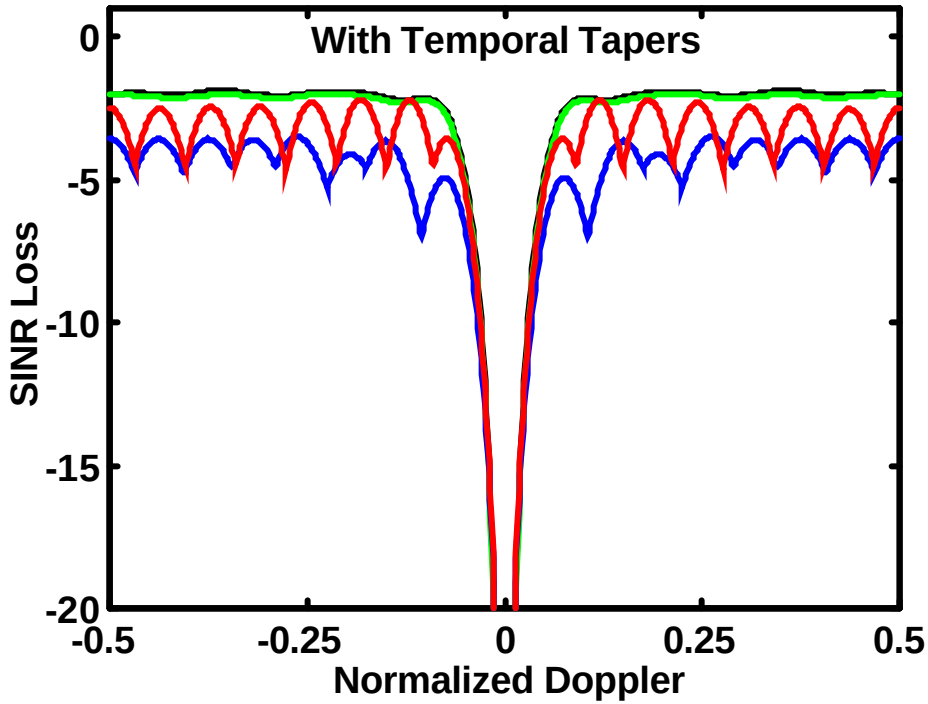
* UFDS = Usable Free Doppler Space (SINR Loss > -5dB)

Baseline STAP Architecture



SINR Loss Comparison of Adjacent Bin and PRI-Staggered STAP





CNR = 60dB

Azimuth = 0º

Number of Pulses = 18

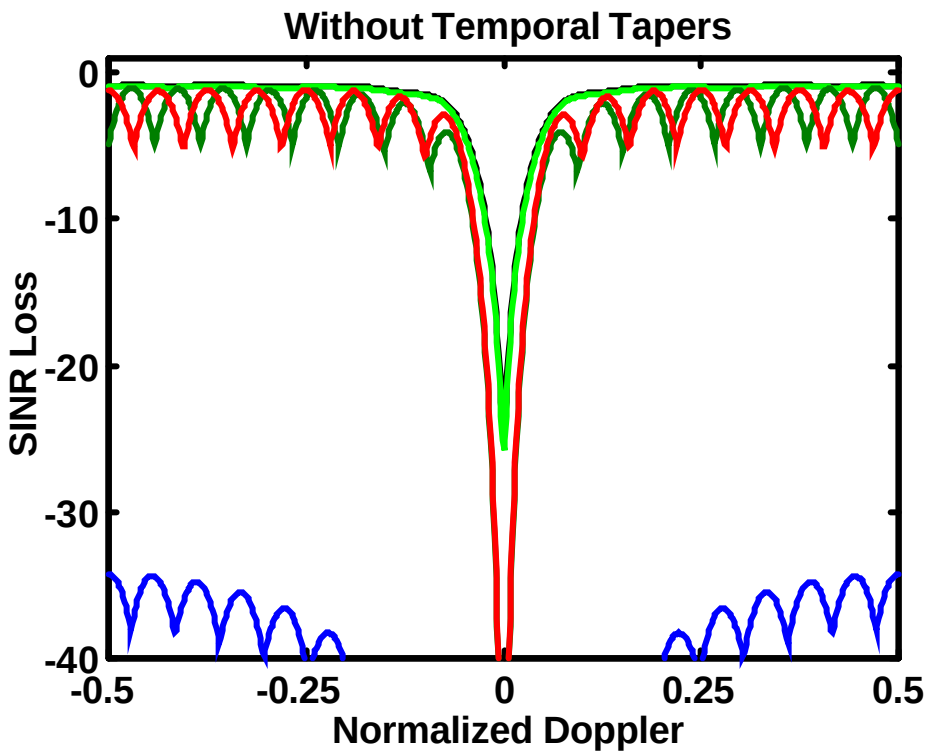
Uniform Clutter Distribution

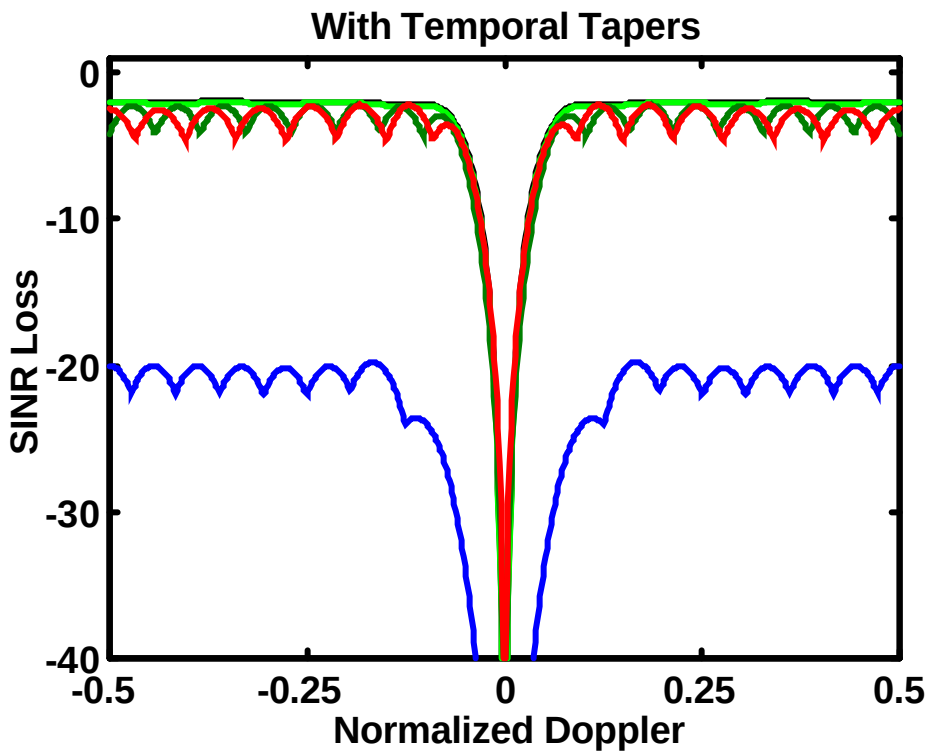
- Fully Adaptive, UFDS = 0.915
- Beamspace K=5, UFDS = 0.912
- Adjacent Bin, UFDS = 0.899
- Pri-Staggered, UFDS = 0.859

- Fully Adaptive, UFDS = 0.902
- Beamspace K = 5, UFDS = 0.899
- Adjacent Bin, UFDS = 0.765
- Pri-Staggered, UFDS = 0.895

Select PRI-Staggered STAP as best compromise of performance, computation complexity and training requirements

Comparison of Different Temporal Dimensions for PRI-Staggered STAP





Uniform Clutter Distribution

Number of Pulses = 18

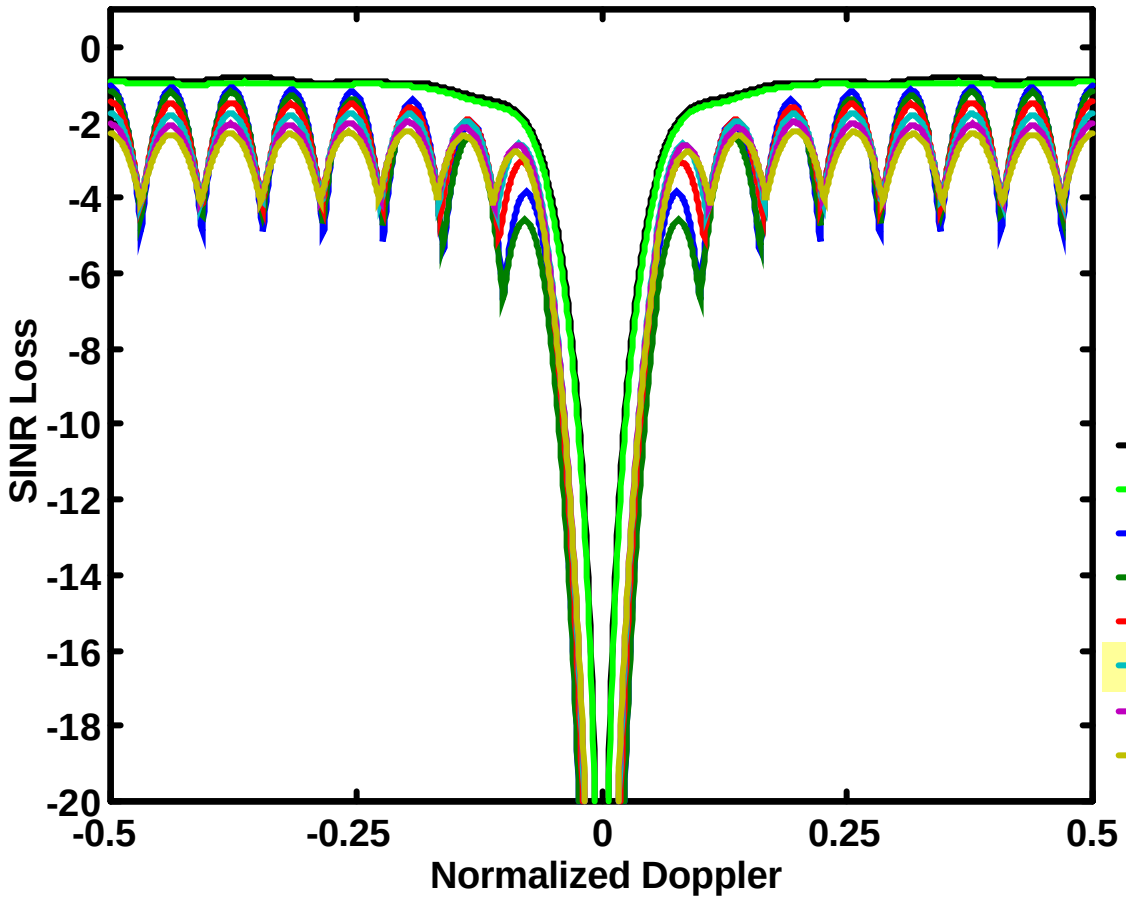
Azimuth = 0º

CNR = 60dB

- Fully Adaptive, UFDS = 0.915
- Beamspace, K=5, UFDS = 0.912
- PRI-Staggered, 1 DFT, UFDS = 0
- PRI-Staggered, 2 DFTs, UFDS = 0.845
- PRI-Staggered, 3 DFTs, UFDS = 0.859

- Fully Adaptive, UFDS = 0.915
- Beamspace, K = 5, UFDS = 0.899
- PRI-Staggered, 1 DFT, UFDS = 0
- PRI-Staggered, 2 DFTs, UFDS = 0.889
- PRI-Staggered, 3 DFTs, UFDS = 0.895

Comparison of Different Temporal Windows for PRI-Staggered STAP



CNR = 60dB
Azimuth = 0º
Number of Pulses = 17
Temporal Degrees of Freedom = 2
Uniform Clutter Distribution

- Fully Adaptive, Uniform, UFDS = 0.913
- Beamspace, Uniform, UFDS = 0.909
- PRI-Staggered Uniform, UFDS = 0.841
- PRI-Staggered, 20dB, UFDS = 0.819
- PRI-Staggered, 25dB, UFDS = 0.883
- PRI-Staggered, 30dB, UFDS = 0.887
- PRI-Staggered, 35dB, UFDS = 0.887
- PRI-Staggered, 40dB, UFDS = 0.883

Alternatively, zero-padding the FFT in the Doppler filter or interpolation between adjacent Doppler bins can be used to reduce scalloping losses

Cheap Windows for Staggered PRI

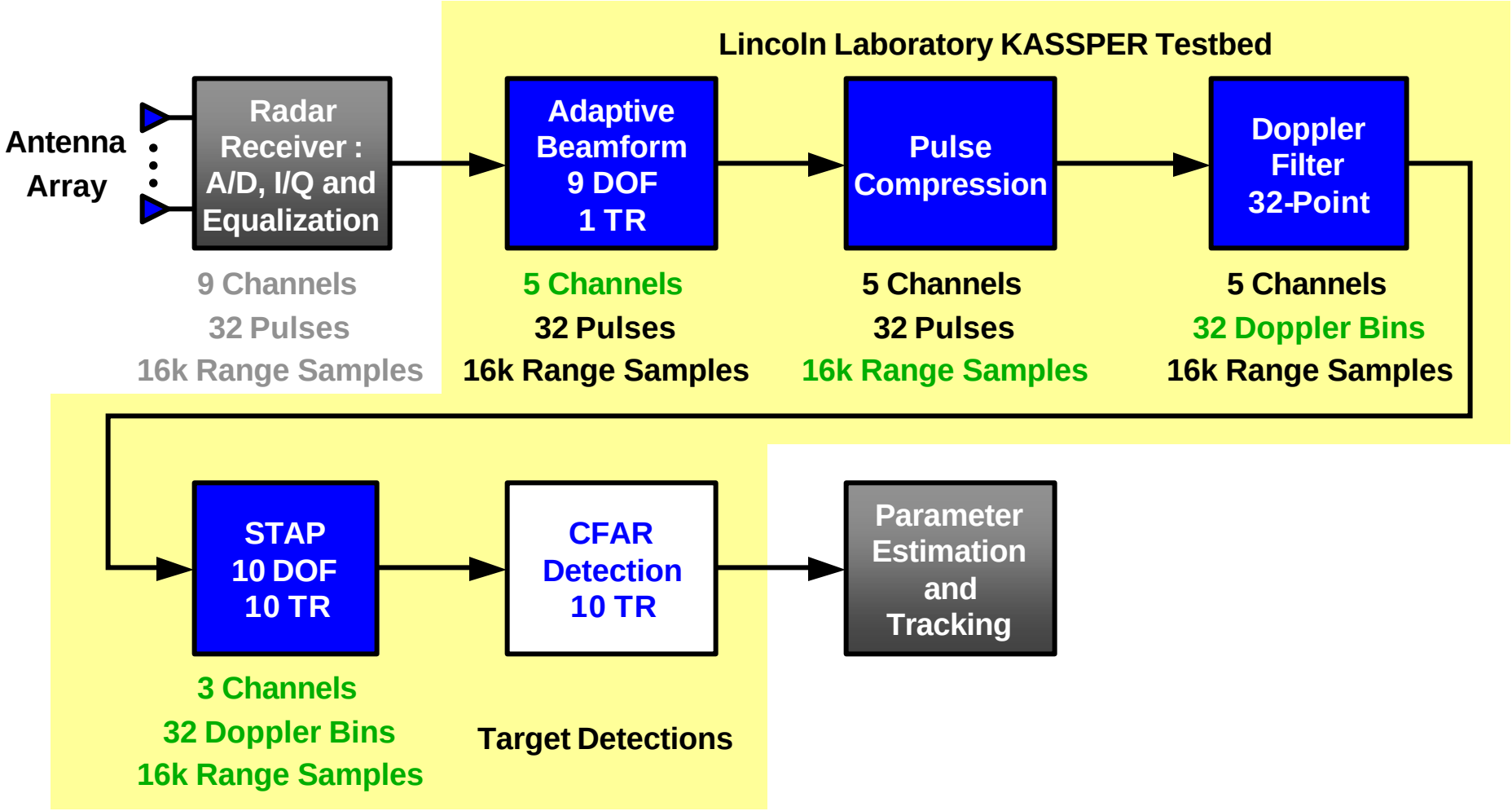
- If no window is used, the second DFT is computed from the first by

$$F_k^{(2)} ? W^k (F_k^{(1)} ? f_K^{(1)} ? f_0)$$

- This trick is not usually available for staggered PRI STAP for windowed DFTs
- But certain windows (Hamming, hanning, and others in a family) can be obtained by 3-point convolution in the frequency domain

$$G_k ? a(F_{k?1} ? F_{k?1}) ? bF_k$$

Baseline STAP Architecture



CFAR Issues

- In baseline system, the CFAR is garden variety split-window moving average.
- All CFAR methods involve a choice of training data appropriate to characterize the interference competing with the sample under test.
- We expect that KASSPER will experiment with a variety of CFAR ideas which choose such training data intelligently.
- The CFAR rule itself may be varied according to whether the particular sample under test is competing with a known clutter discrete.
- Therefore the CFAR in the baseline system should be viewed as a ``place-holder''.

Outline

- Introduction
- Baseline STAP Algorithm Description
- Adequate Gain Constraints
- KASSPER Algorithms
- Summary

Adequate Gain Constraint

- Classical Interference Cancellation uses $w = R^{-1}v$
- This protects adapted gain at one constraint v

- We can protect the adapted gain at multiple constraints $v_1, v_2, \ldots, v_L$ but at a cost
  - Degrees of freedom
  - Cancellation achieved

- We think it is better to constrain adapted gain using inequality constraints

- Choose w to minimize $w^h R w$ subject to $|v_i^h w| > 1$

Algorithm for Adequate Constraints Nulling

$$\min\ w^h R w \quad \text{subject to} \quad |w^h v_i| ? 1$$

$$u ? \begin{bmatrix} ?\mathrm{Re}(w)? \\ ?\mathrm{Im}(w)? \end{bmatrix} \qquad ? ? \begin{bmatrix} ?\mathrm{Re}(R) & ?\mathrm{Im}(R)? \\ ?\mathrm{Im}(R) & \mathrm{Re}(R)? \end{bmatrix}$$

$$c_i ? \begin{bmatrix} ?\mathrm{Re}(v_i)? \\ ?\mathrm{Im}(v_i)? \end{bmatrix}$$

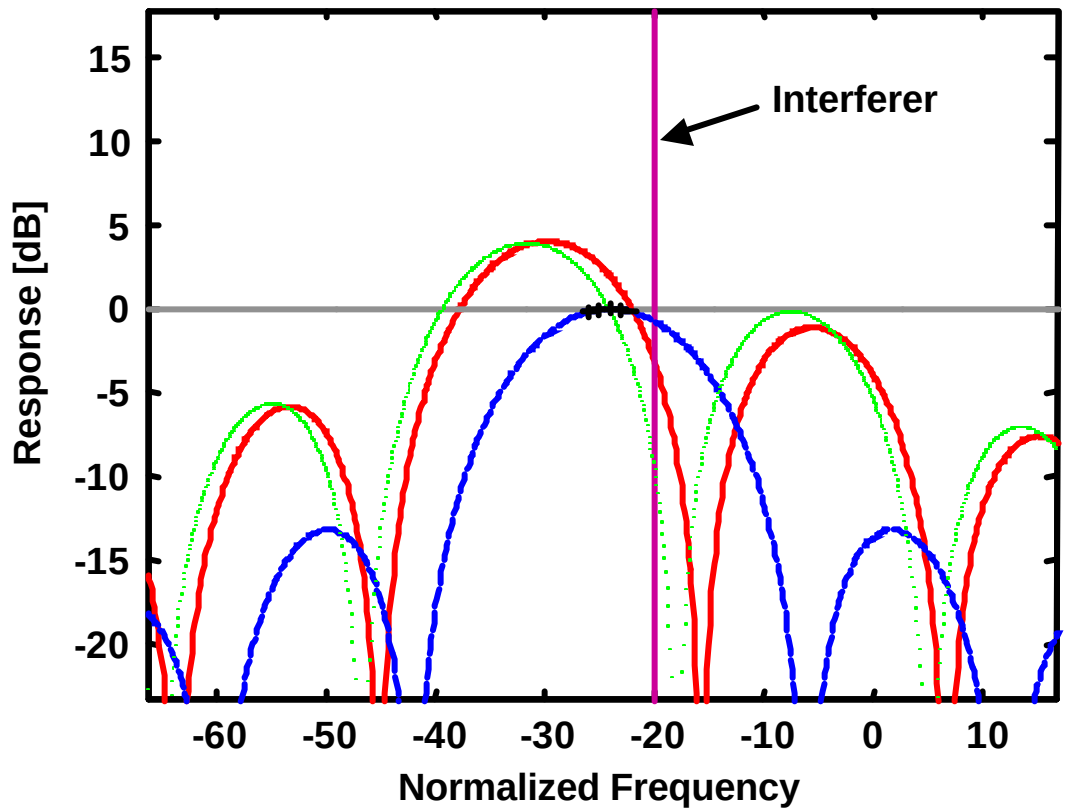
CANNED PROGRAM

min $u^t P u$ subj to $c_i^t u ? 1$

$\xrightarrow{u}$ $v_i ? v_i \exp(? j \arg(w^h v_i))$

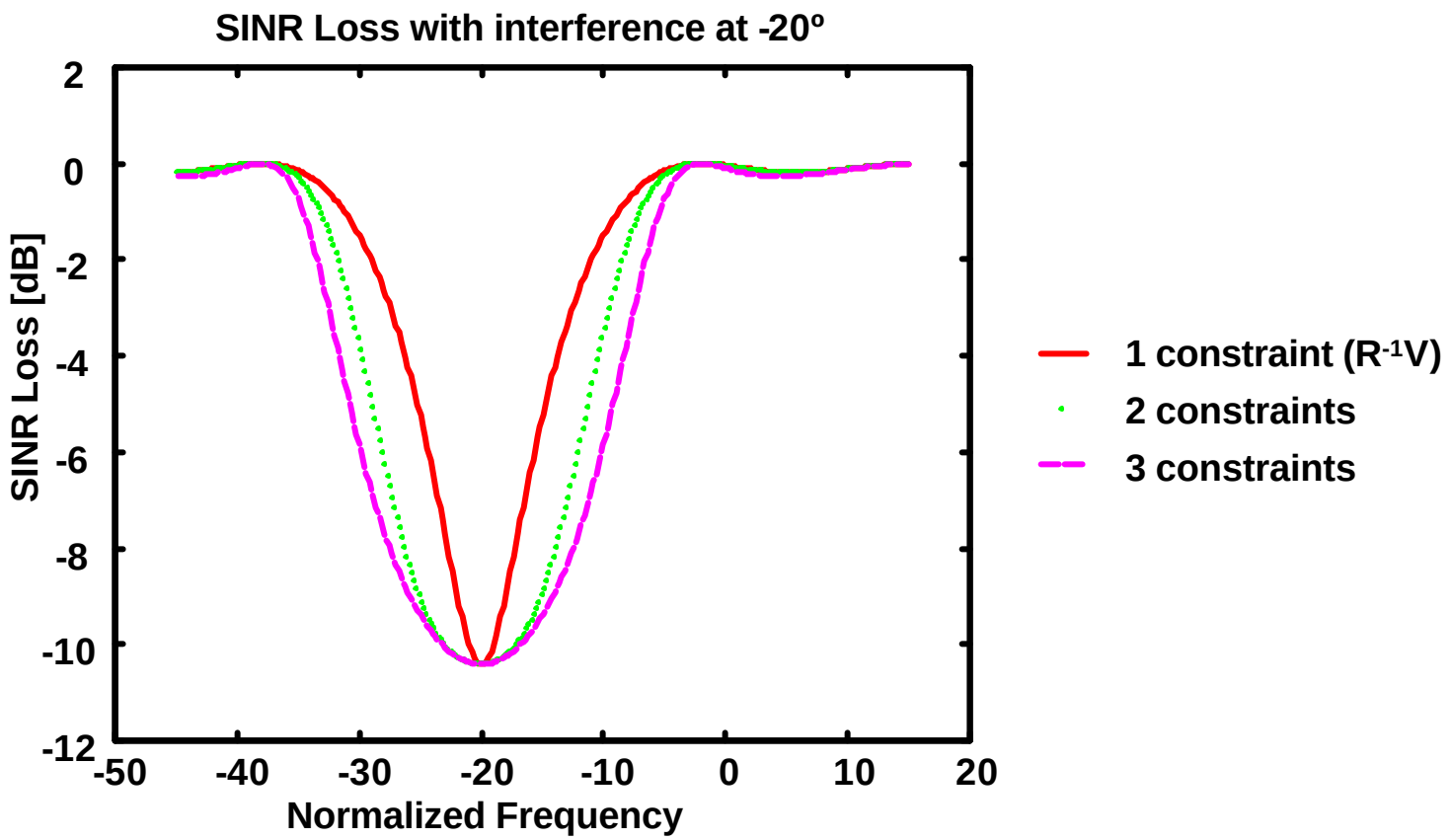
Convergence is rapid

20-tap FIR Filter Example
Adequate Gain Constraint vs MGC



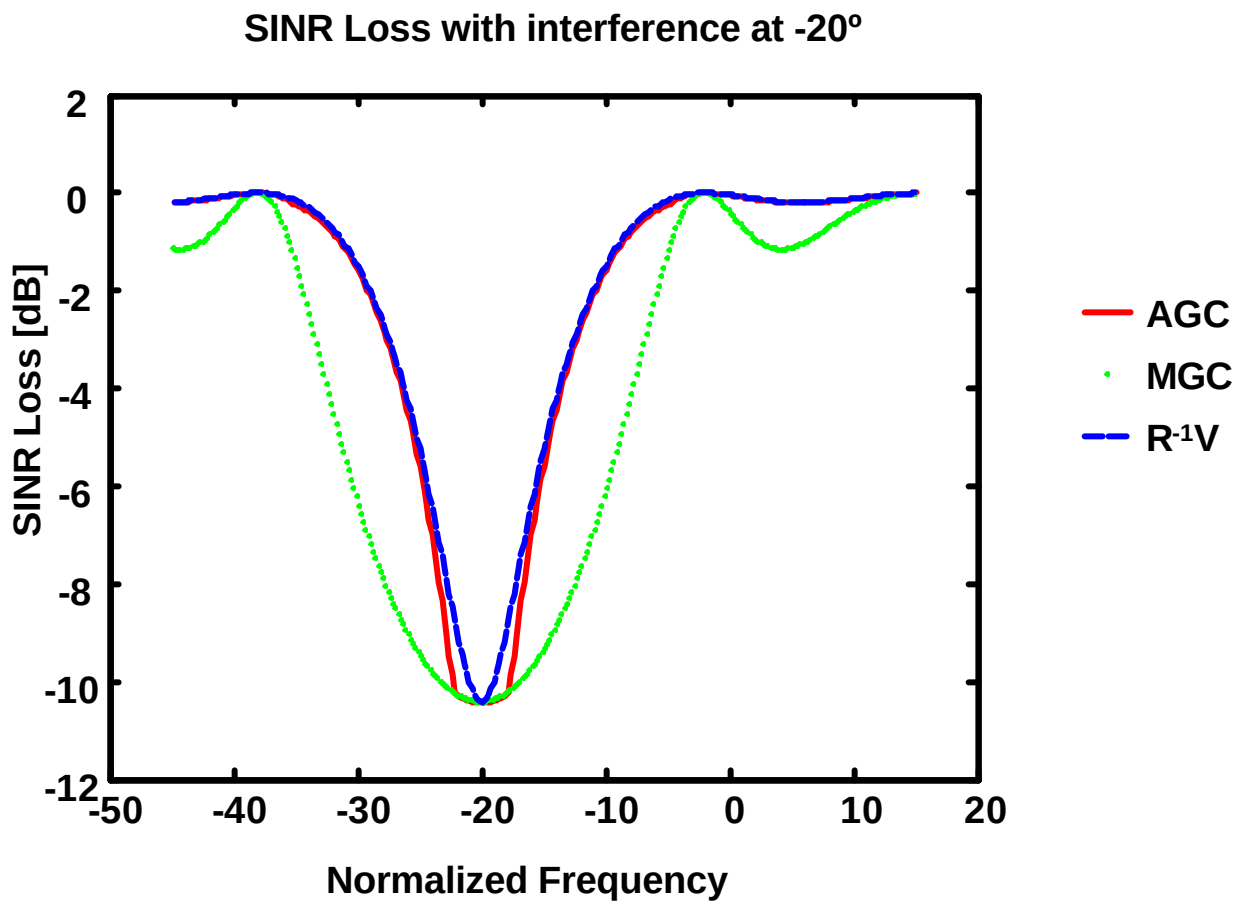
- Single Constraint ($w = R^{-1}v$)
- Multiple Linear Constraints (at five positions marked '+')
- Five Adequate Gain Constraints

SINR Loss vs Number of Multiple Gain constraints



- Adapted SINR is poor near the jammer
- Interference notch is narrowest for $R^{-1}v$
- Interference notch widens dramatically with additional constraints

SINR Loss for Adequate Gain Constraint



Constraints span same 5 degrees as MGC method

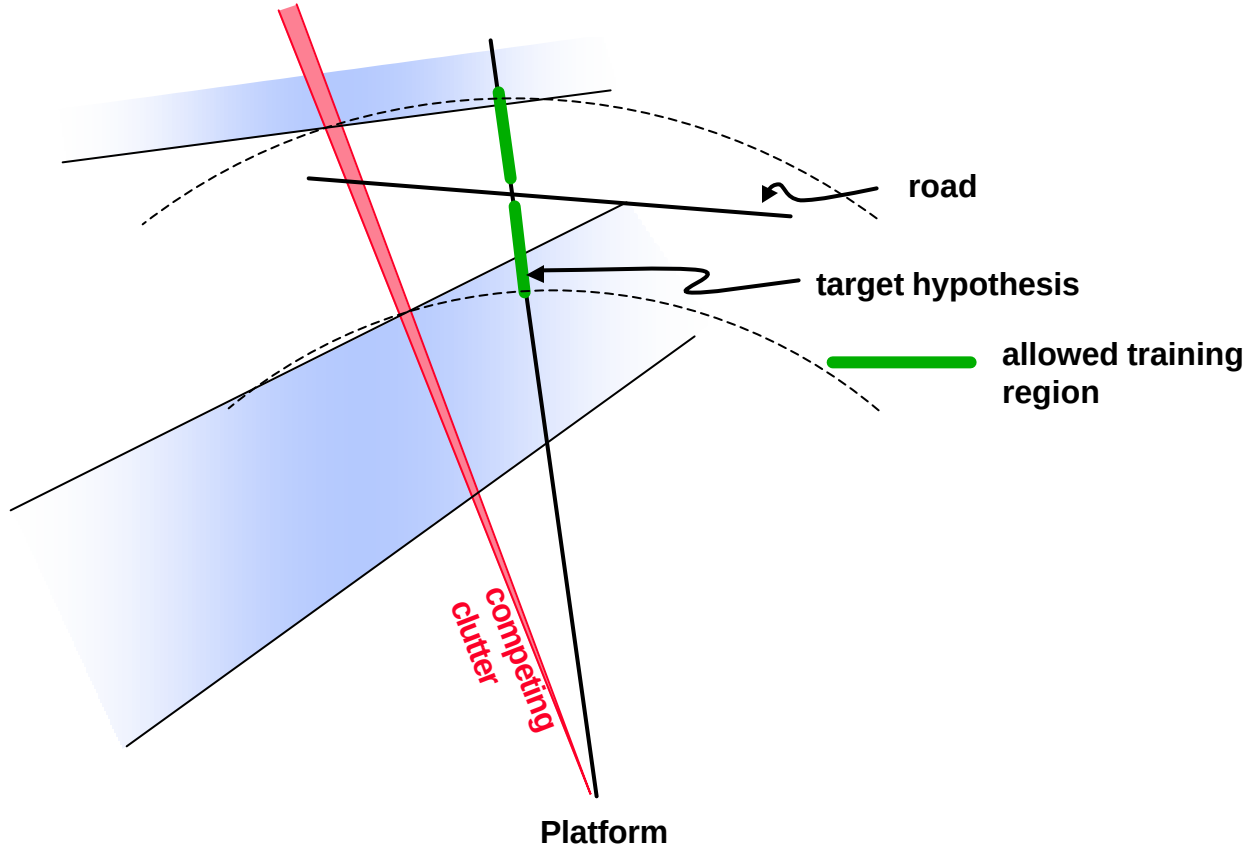
Outline

- Introduction
- Baseline STAP Algorithm Description
- Adequate Gain Constraints
- KASSPER Algorithms
- Summary

Terrain Selective Training

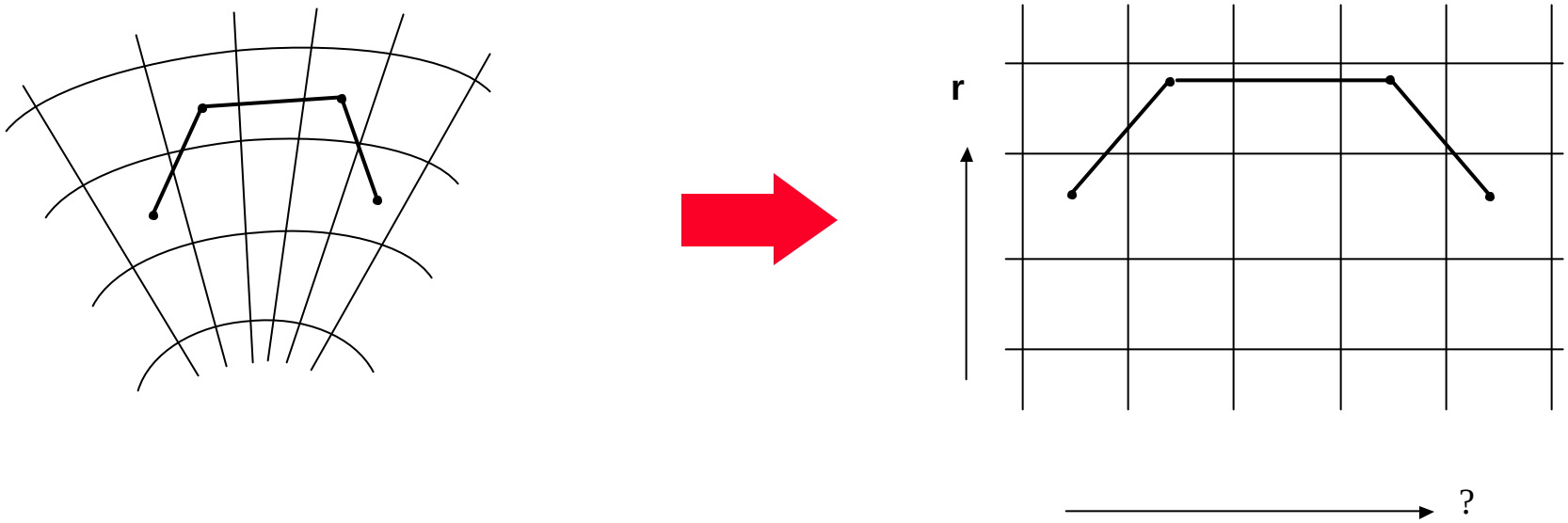
STAP training

- **use ranges with clutter from same terrain type**
- **exclude ranges with roads in look direction**



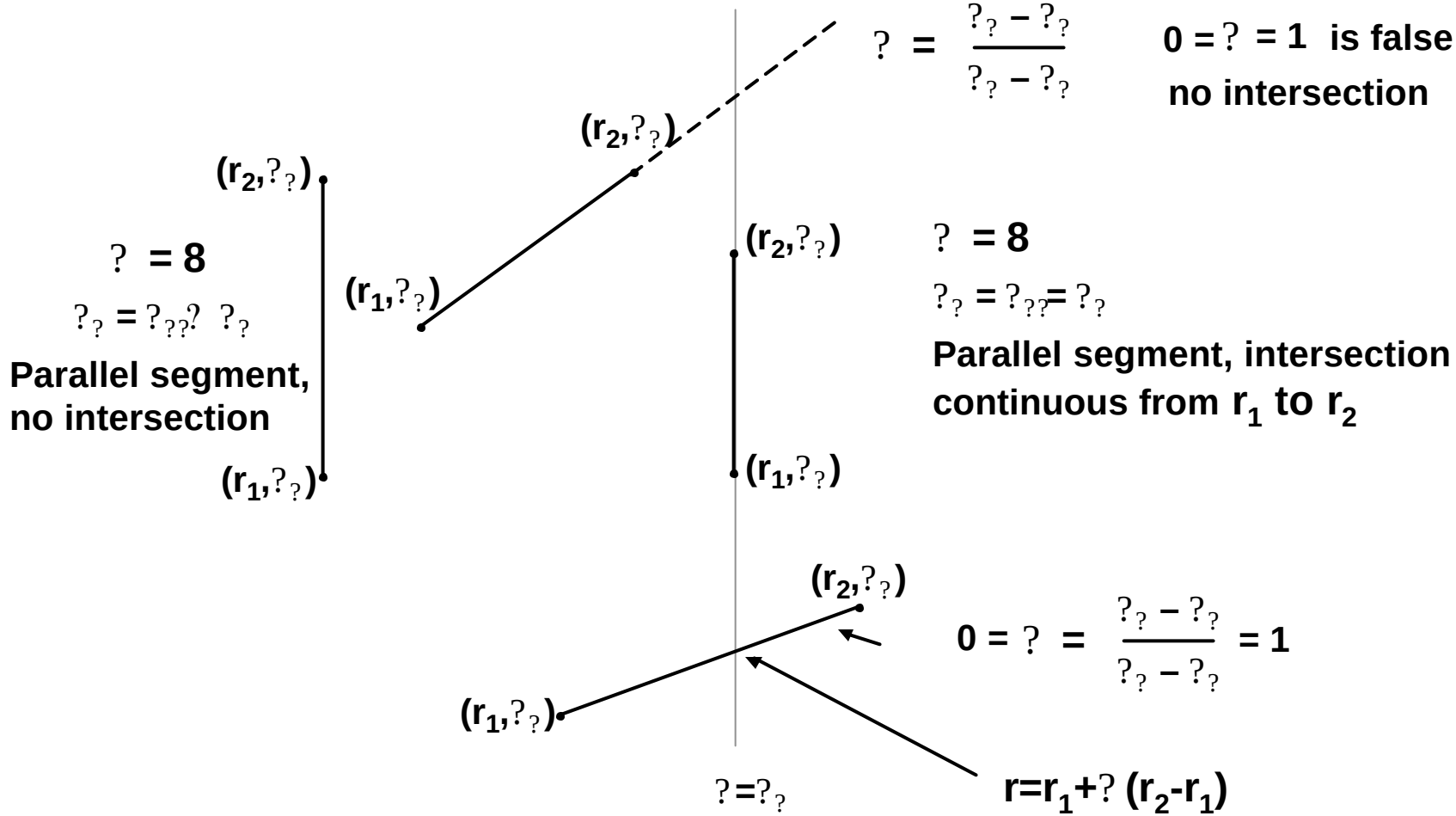
The training range is different for each target doppler hypothesis

Terrain Boundaries and Roads



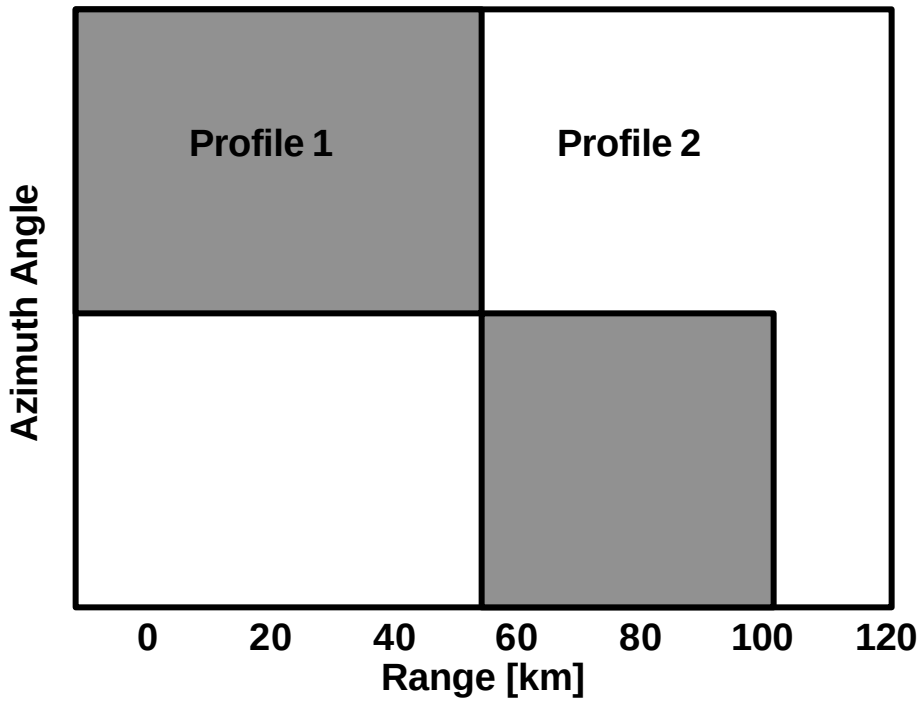
Roads and terrain boundaries composed of straight line segments.
Line segment end points (r_i, ?$_i$) treated as rectangular coordinates.

Intersection of Line Segments with 'Constant Theta' Lines



LL STAP Simulation

Clutter Distribution



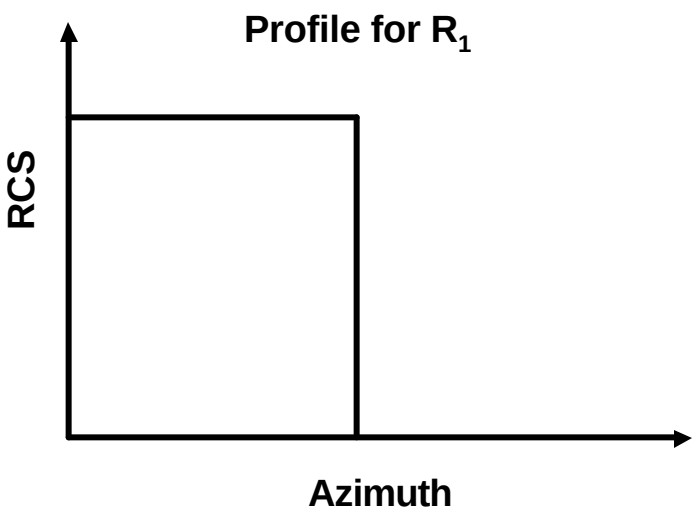
Synthesized Targets

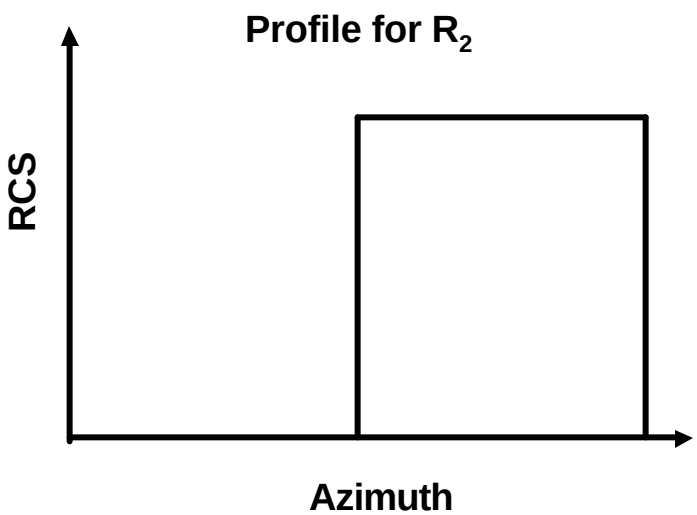
Range [km]	Velocity [m/sec]	SNR [dB]
70.0	-2.2	25
70.4	+3.5	25

SINRLoss Expressions for Terrain Type Mismatches

$$w(R) \,?\, R^{?1} v$$

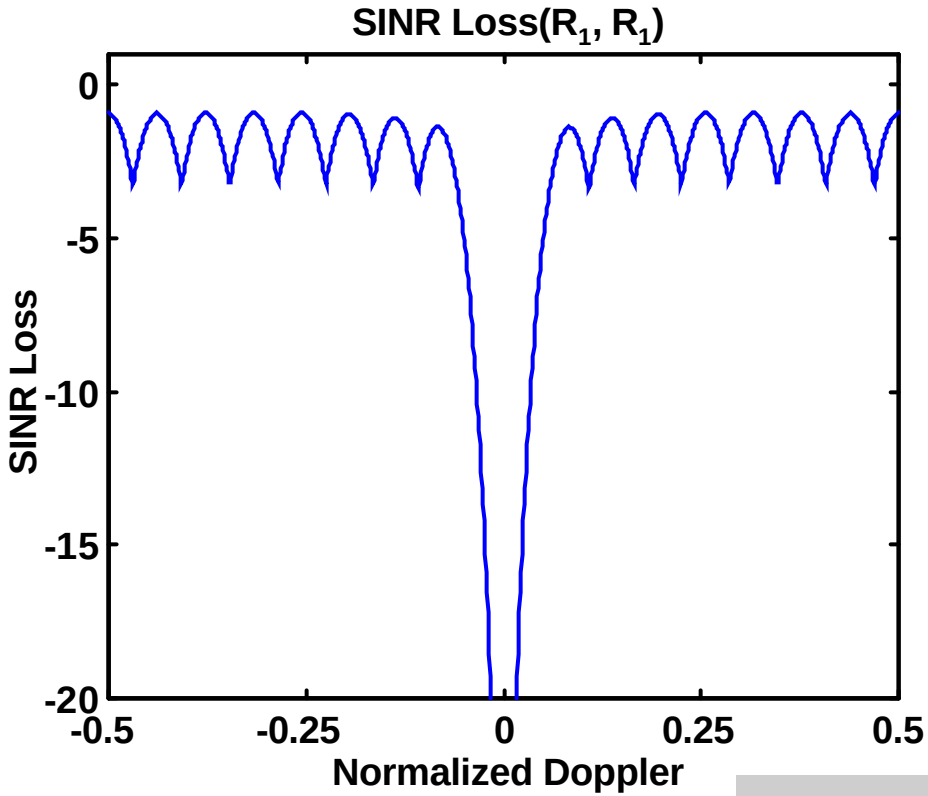
$$\mathrm{SINRLoss}(R_1,R_2) \,?\, \frac{\left|w(R_1)^H v\right|^2}{w(R_1)^H R_2(R_1)} \,?\, \frac{v^H R_{noise} v}{\left|v^H v\right|^2}$$

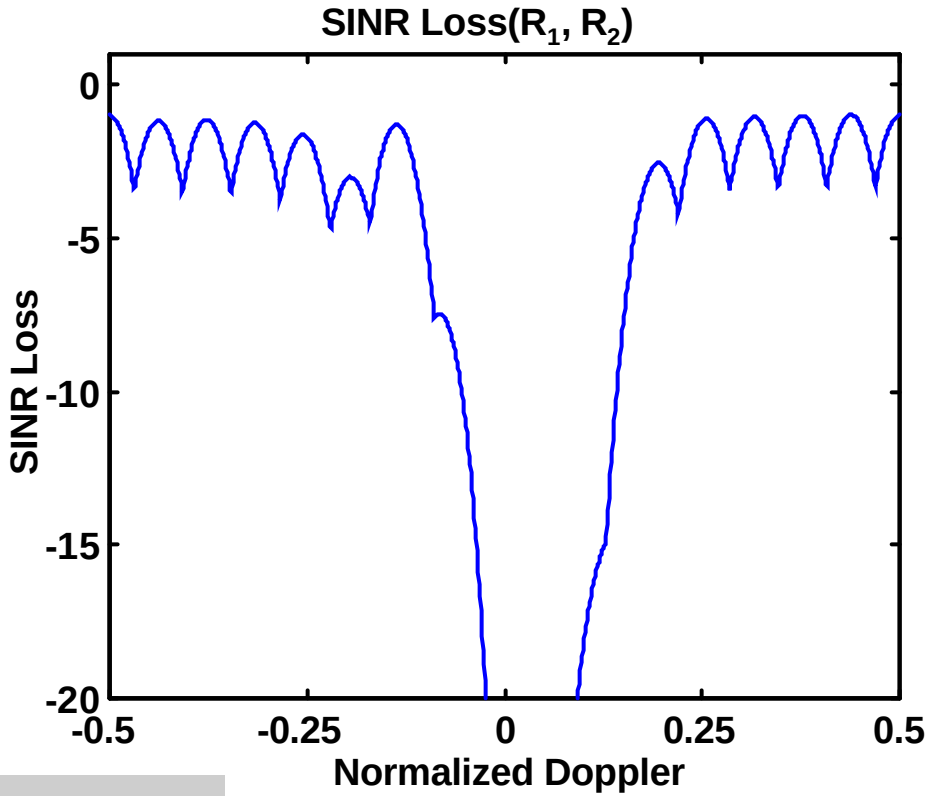




STAP Comparison

for Mismatched Terrain Types



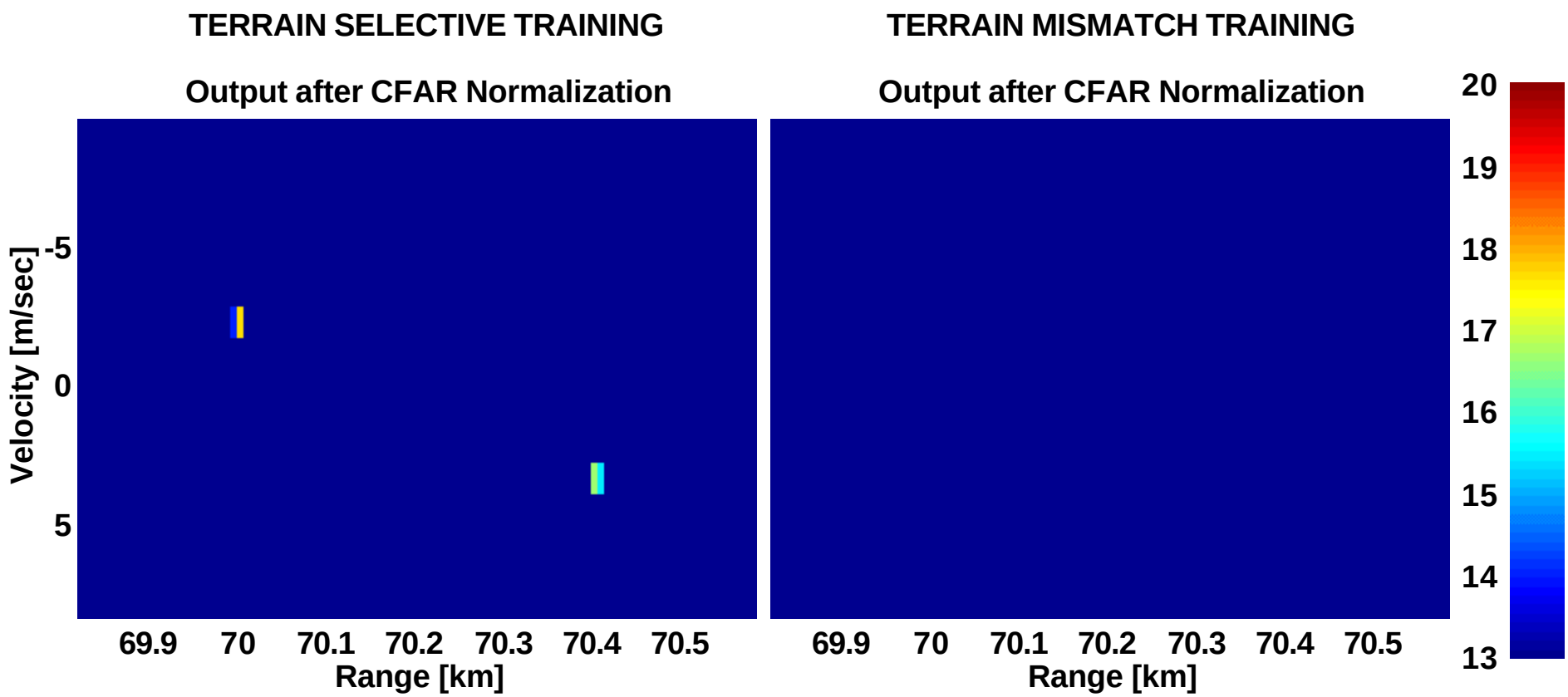


PRI-Staggered, UFDS = 0.899

PARAMETERS	
CNR	40
Spatial DOFs	5
Temporal DOFs	2
Pulses	17
Azimuth	0º

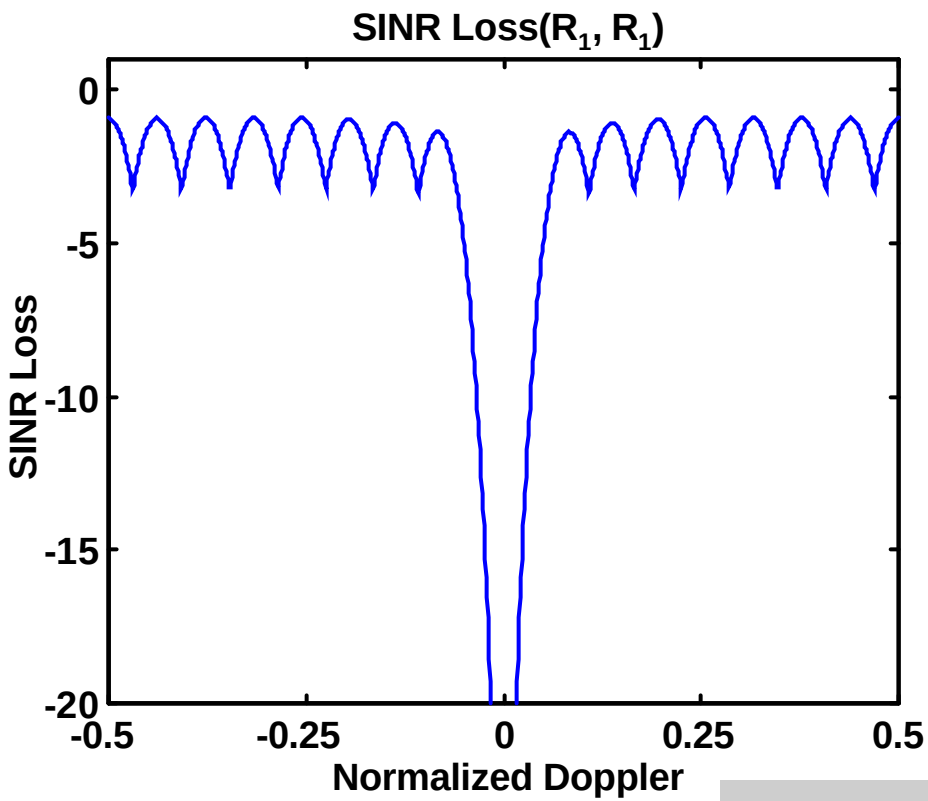
PRI-Staggered, UFDS = 0.737

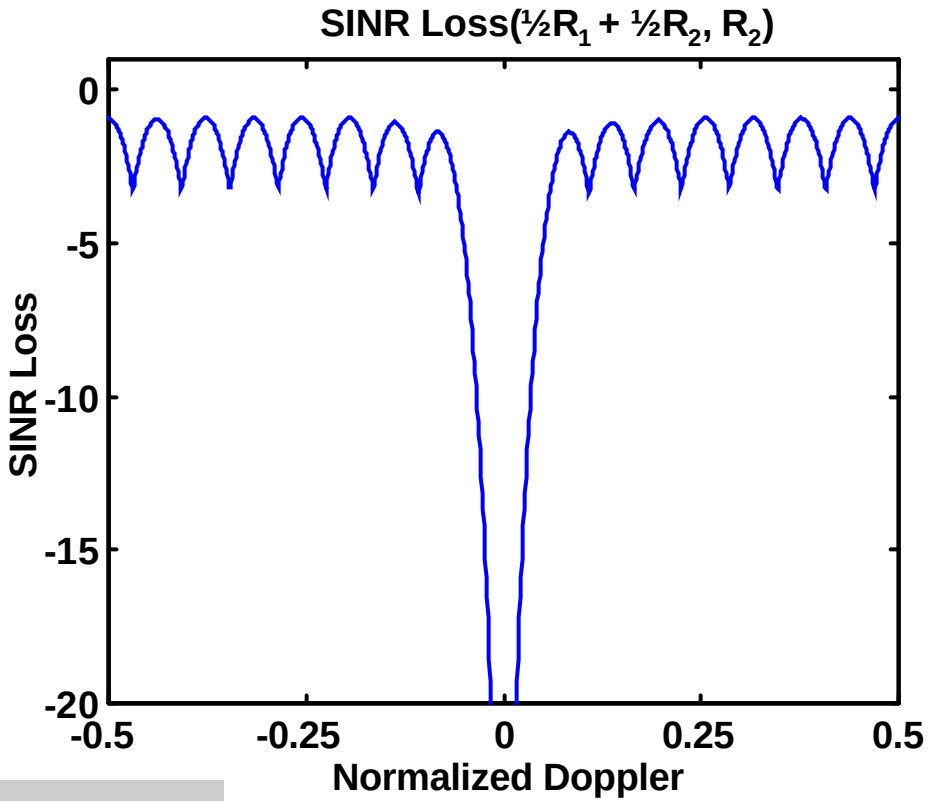
Simulator Example for Terrain Mismatch Terrain Training



STAP Comparison

for Mismatched Terrain Types



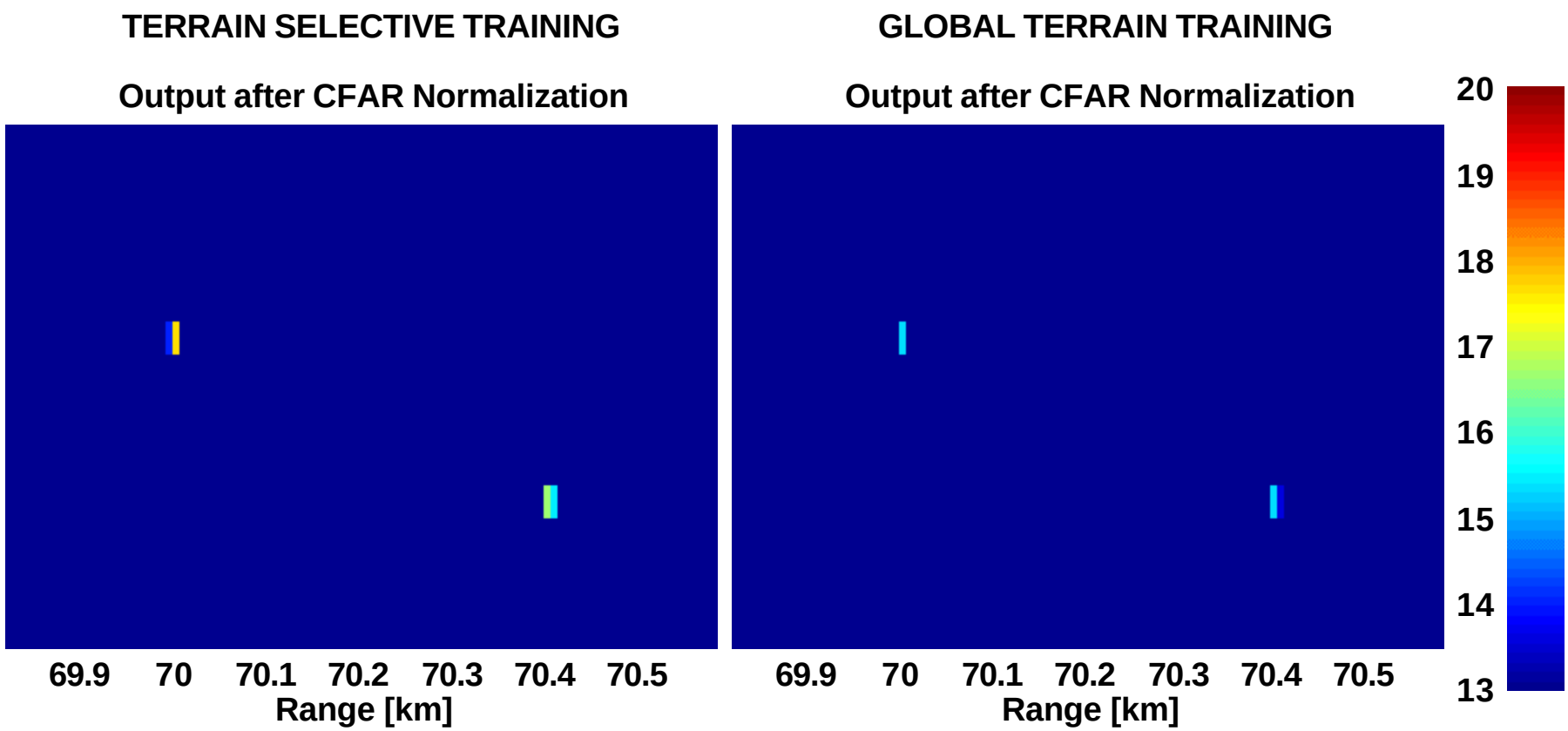


PRI-Staggered, UFDS = 0.899

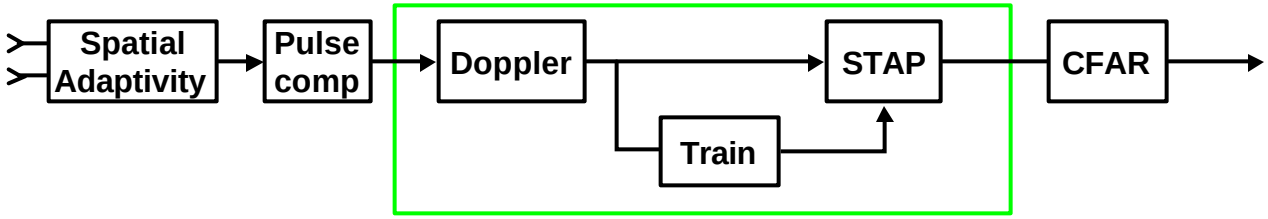
PARAMETERS	
CNR	40
Spatial DOFs	5
Temporal DOFs	2
Pulses	17
Azimuth	0º

PRI-Staggered, UFDS = 0.899

Simulator Example for Terrain Mismatch Global Training



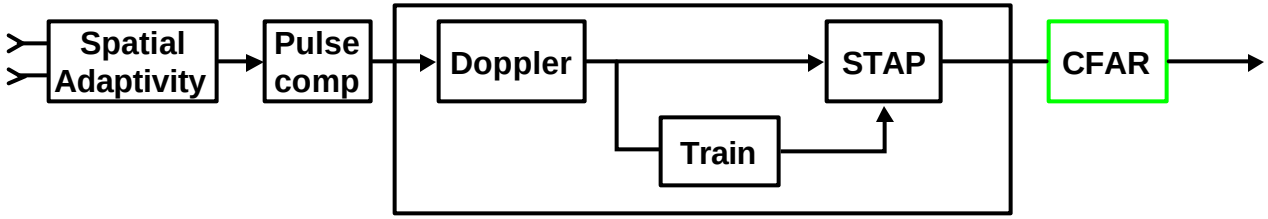
KASSPER Algorithm Taxonomy



STAP METHODS

- Obtain prior knowledge from maps, SAR, other sensors, previous looks.
- Train using terrain knowledge.
- Train using data-adaptive selection or exclusion
- Edit detections out of training and try again
- Protect target gain with multiple adequate gain constraints.
- Special treatment of clutter discretes
  - Suppress known clutter discretes with artificial training.
  - Exclude known clutter discretes from global training.
- Use a-priori estimated correlation to reduce training requirements.
- Apply STAP in stressed doppler bins, easier processing in other doppler bins

KASSPER Algorithm Taxonomy



CFAR METHODS

- **Obtain prior knowledge from maps, SAR, other sensors, previous looks.**
- **Train using terrain knowledge.**
- **Edit detections out of training and try again**
- **Adjust threshold using a-priori expectations, e.g. targets more likely on roads, less likely on steep terrain.**
- **Mark detections `doubtful' near known clutter discretes.**
- **Use multi-dimensional tests (like ACE) to distinguish target-in-clutter from known-discrete-in-clutter.**

Summary

- Designed and specified a baseline STAP architecture for the Lincoln Laboratory KASSPER Testbed to
  - Evaluate candidate KASSPER algorithms
  - Benchmark real-time processing
- Initiated performance assessment of KASSPER algorithms for terrain specific training
- Developed a novel adaptive beamforming technique
  - Initial results indicate adequate gain constraints provide benefits over multiple linear constraints
- Outlining other candidate KASSPER algorithms for consideration