Footprint reduction by angle-weighted stacking after migration

Joanna K. Cooper, Gary F. Margrave,
and Don C. Lawton

November 21, 2008

CREWES

UNIVERSITY OF CALGARY

MO SHÙILE TOGAM SUAS

Outline

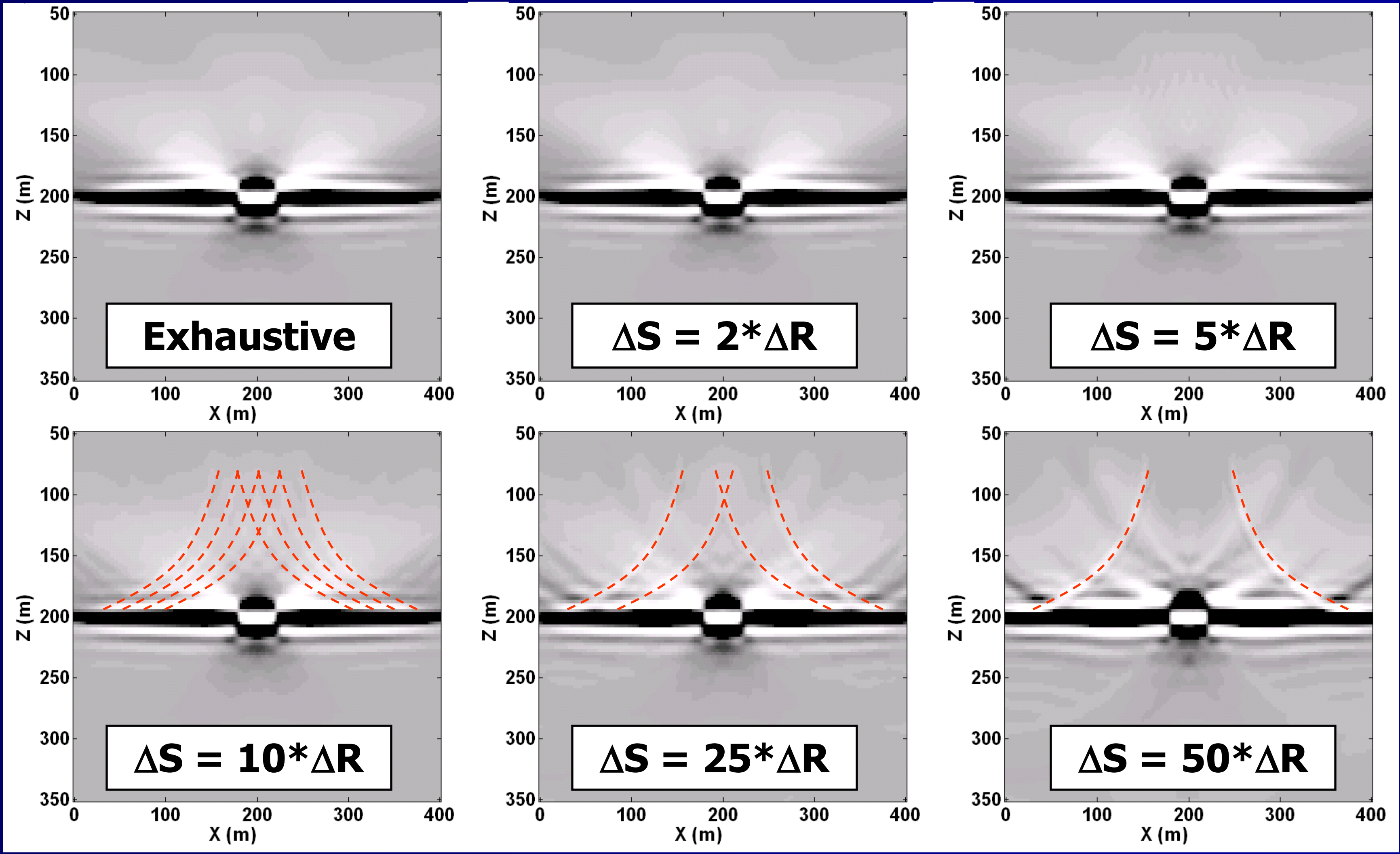
- Review of 2007 footprint simulations
- Description of method for angle-weighted stacking
- Application of method in 2D
- Illustration of method in 3D
- Conclusions and future work

Recap: 2D Footprint Simulations

- Modelled an exhaustive 2D dataset: shots and receivers spaced at 5 m intervals over a 400 m long model
- Created five shot decimations with shot spacings of 10 m, 25 m, 50 m, 100 m, and 200 m
- Applied Kirchhoff prestack migration and stacked migrated shot records

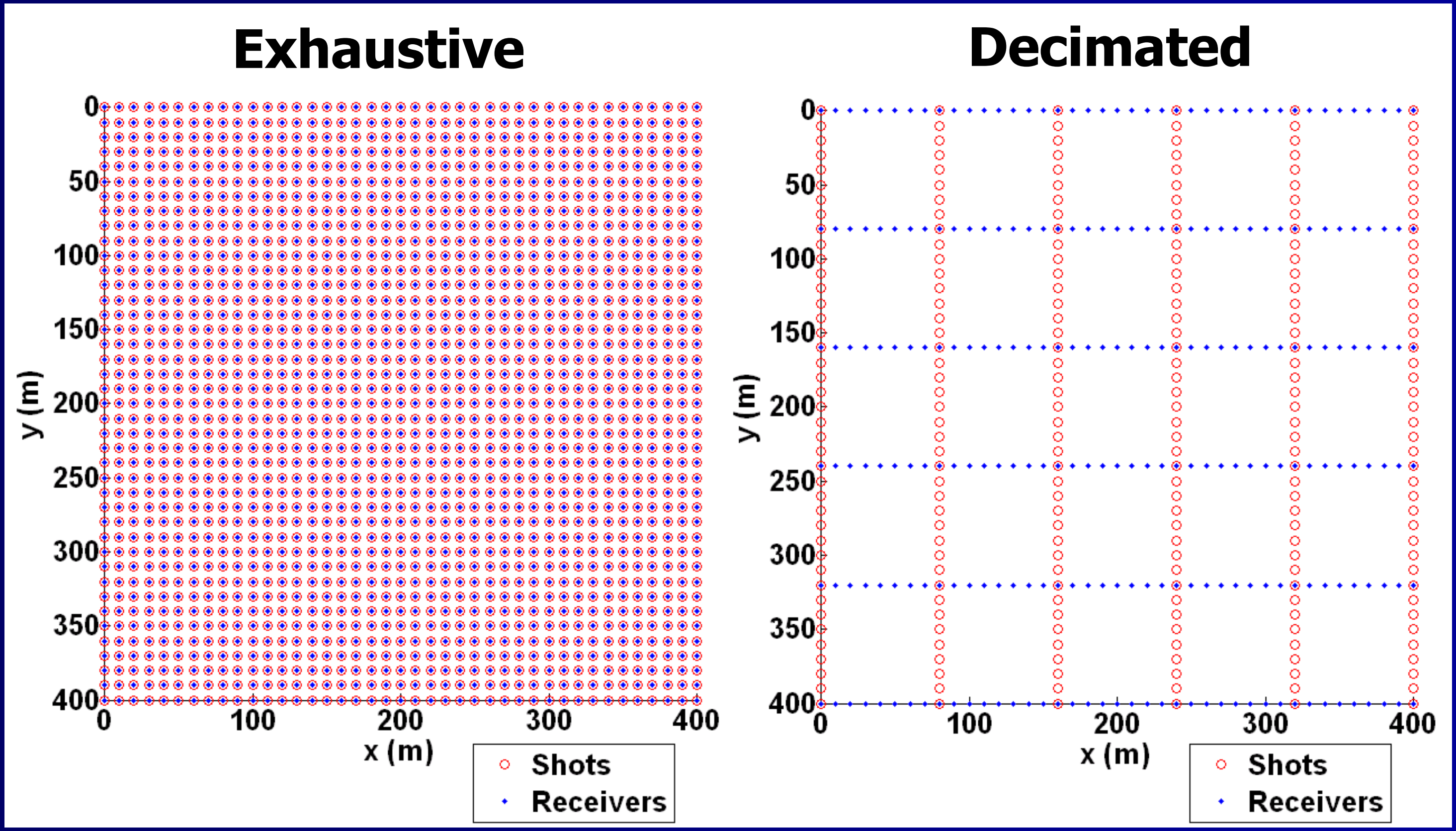
Recap: 2D Simulation (after Cary, 2007)

- Prestack migrated sections:



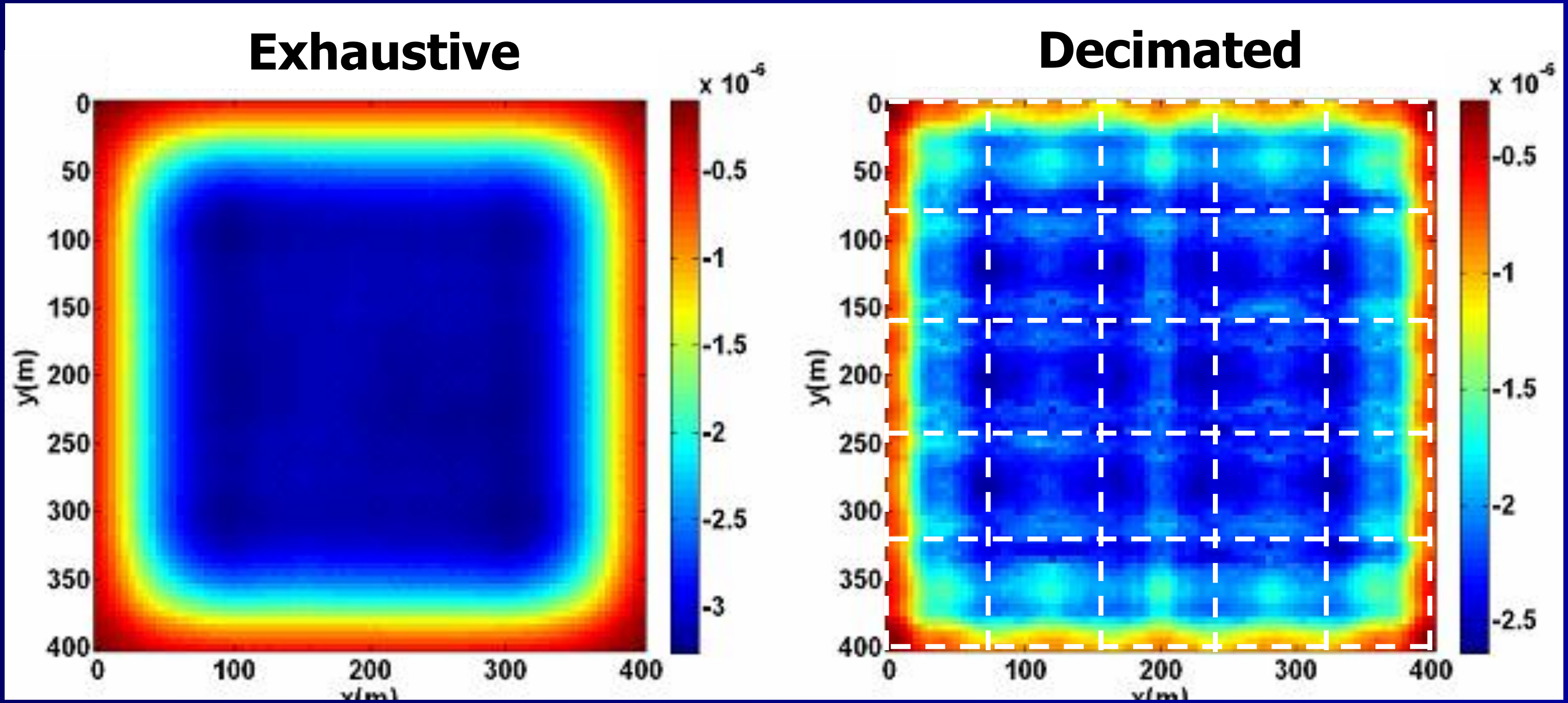
Recap: 3D Footprint Simulations

- Modelled an exhaustive dataset via Rayleigh-Sommerfeld and created one decimation
- Migrated with 3 prestack migration algorithms



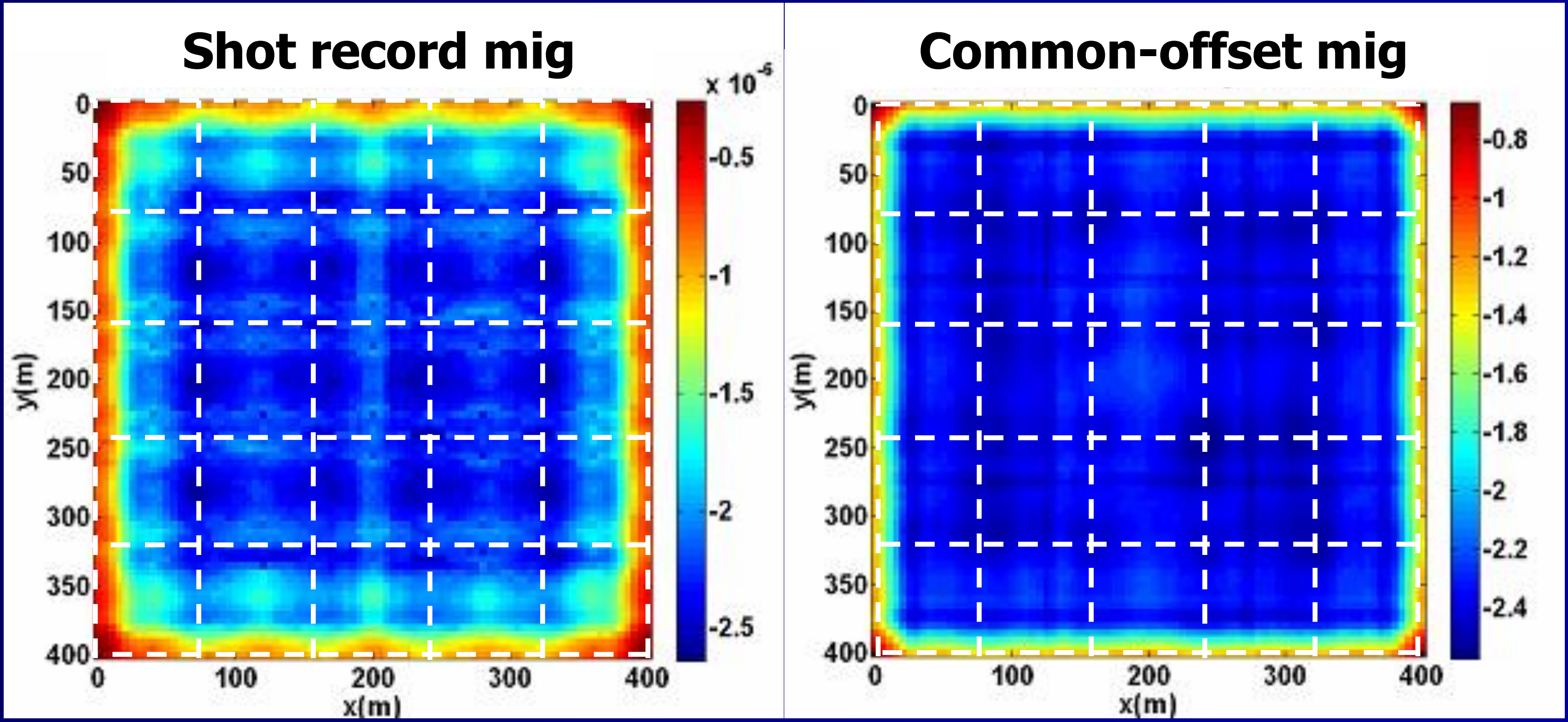
Recap: 3D Footprint Simulations

- Comparison: exhaustive vs. decimated on a featureless reflector



Recap: 3D Footprint Simulations

- Comparison of different migration algorithms for the decimated dataset:



Recap: ’07 Footprint Simulations

- 2D: Footprint manifests as residual migration wavefronts in decimated datasets
- 3D: Periodic amplitude variations appear in migrated depth slices
- 3D: Migration algorithms, in particular migration weights, make a big difference in observed footprint

➔ Can footprint reduction be achieved via prestack migration weights?

Method

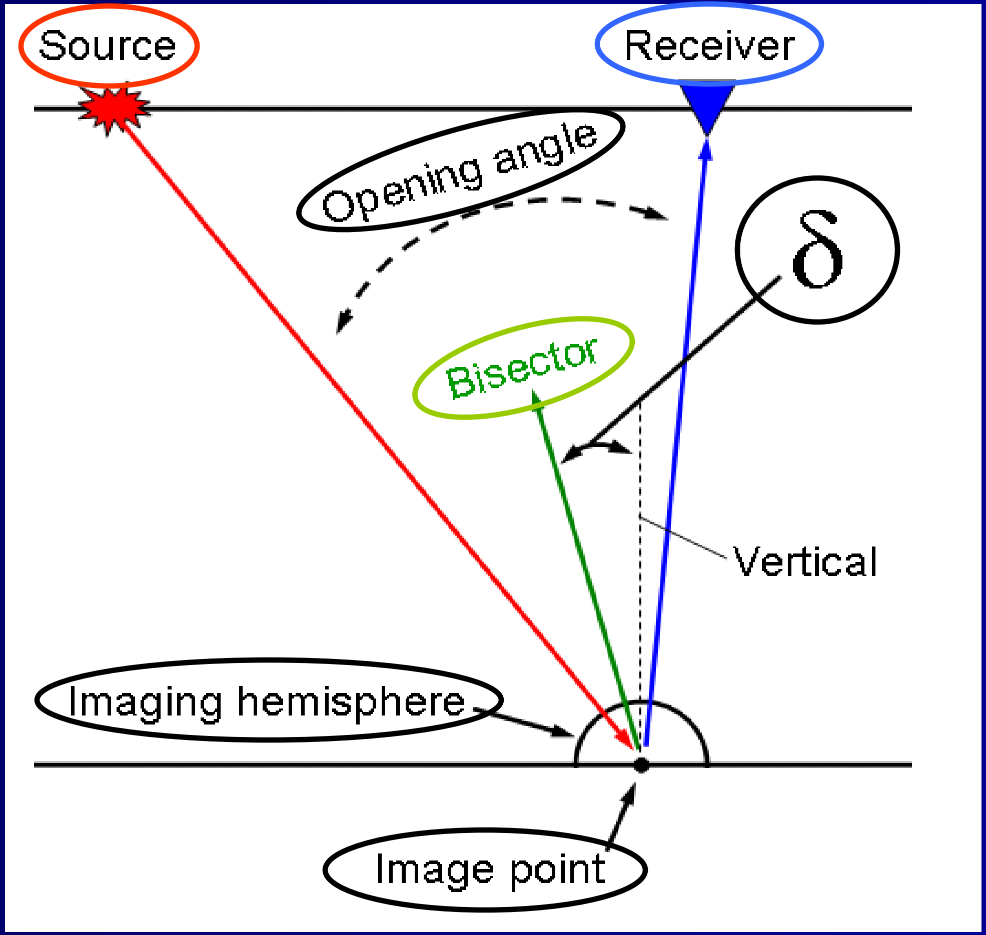
- Bleistein migration weights convert from uniform, infinite source and receiver coverage to uniform angular illumination of image point
- Still need to compensate for discrete, finite, irregular sampling (e.g. decimated dataset)
- Normalization may allow wavefronts to properly interfere

Method

- Analogy: numerical integration
- $\int f(x)dx \approx \Delta x \sum_j f(x_j)$, only if samples are regular and infinite
- For irregular sampling, must compute a weighted sum: $\sum_j f(x_j)\Delta x_j$
- Kirchhoff migration: multidimensional integral in space, approximated by a sum, and weighted in order to achieve uniform illumination of the image point

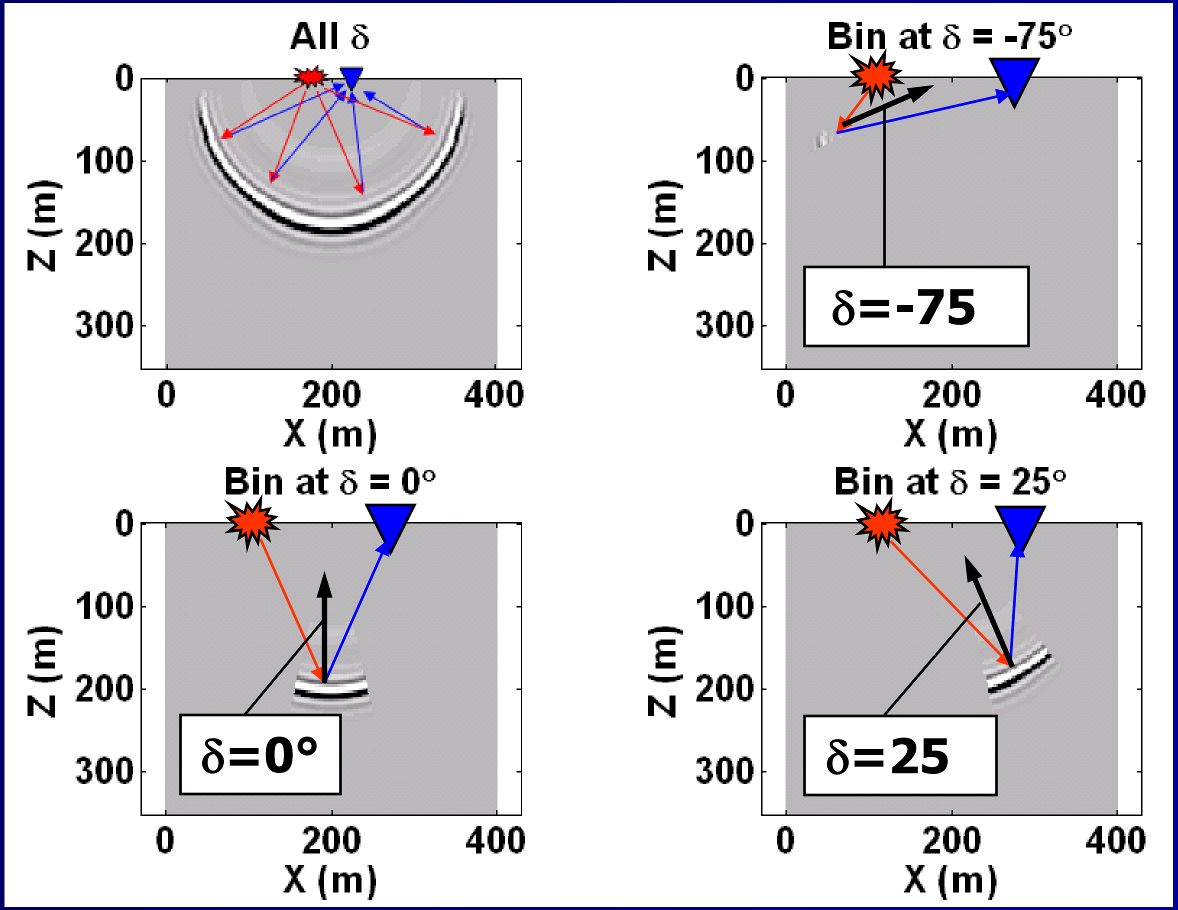
Method

- Concept: illumination of imaging hemisphere by delta angles



Method

- Delta is also the normal to the migration impulse response

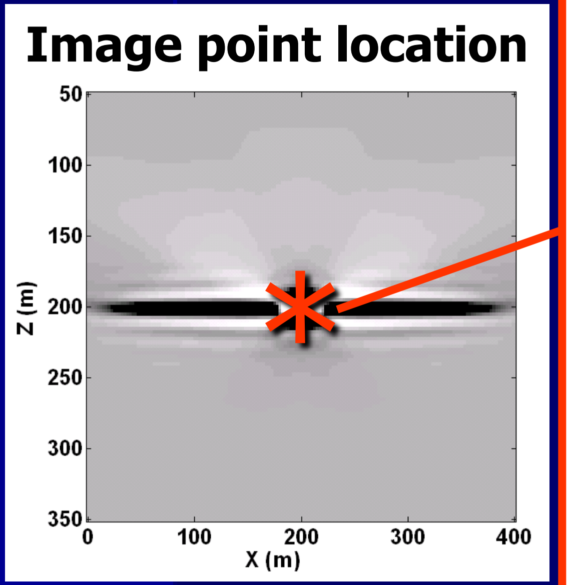


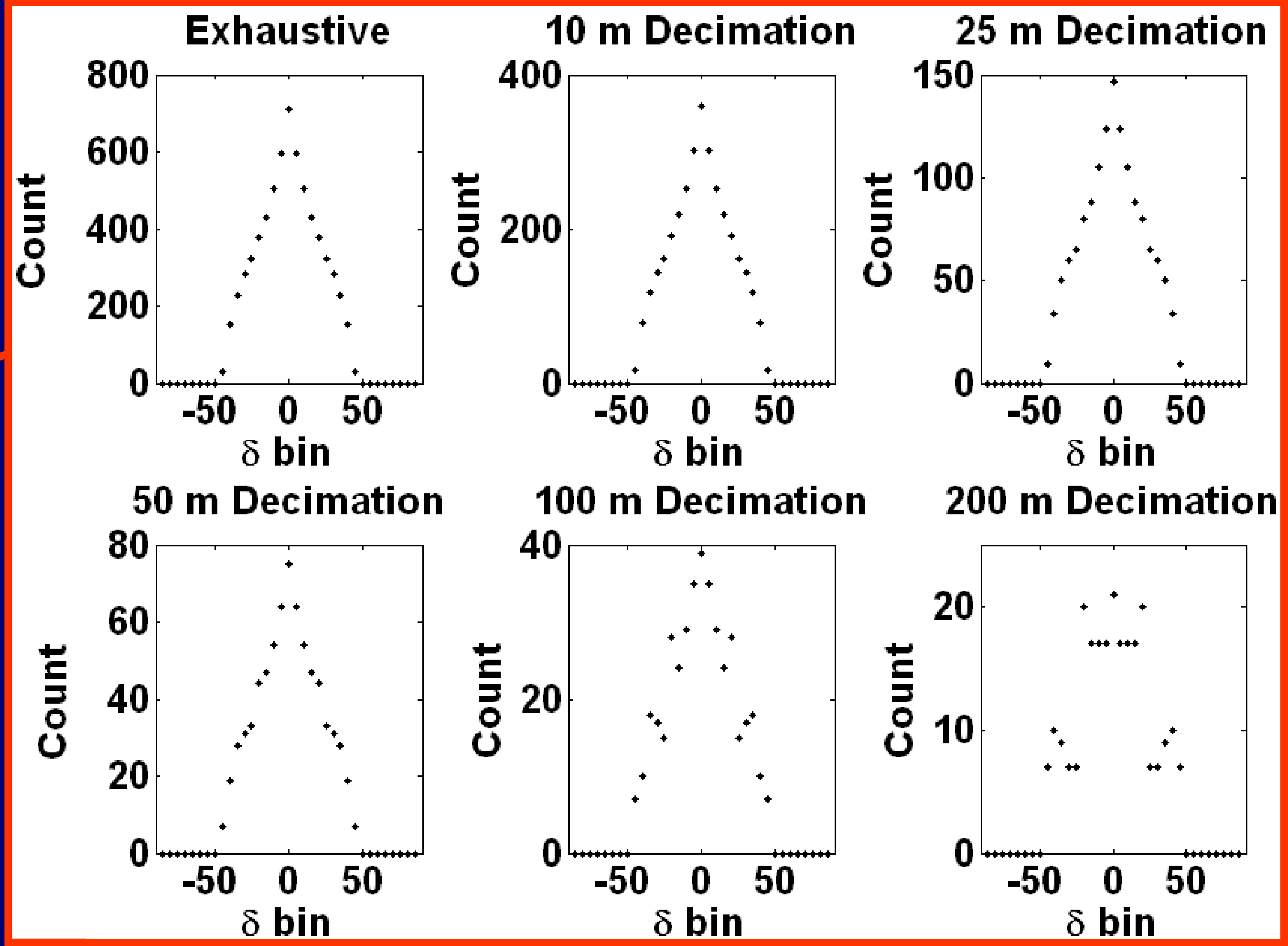
Method

- Consider illumination of imaging hemisphere by delta vectors
- Each source-receiver pair defines a delta angle for each image point
- Want to achieve uniform illumination by normalizing by delta hit counts

Method

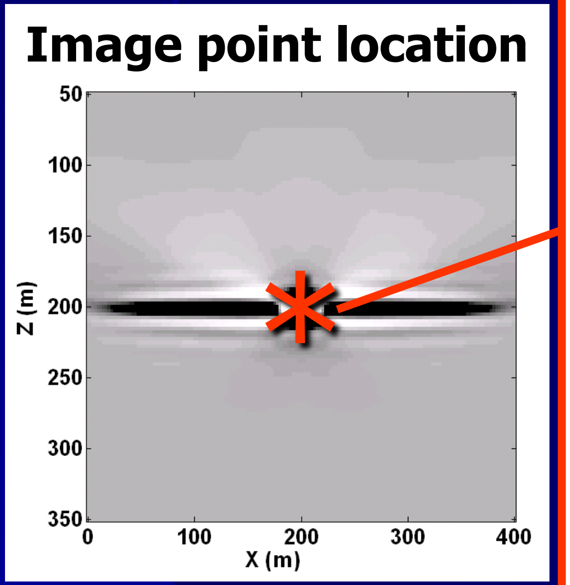
- Delta bin hit counts vs. shot decimation

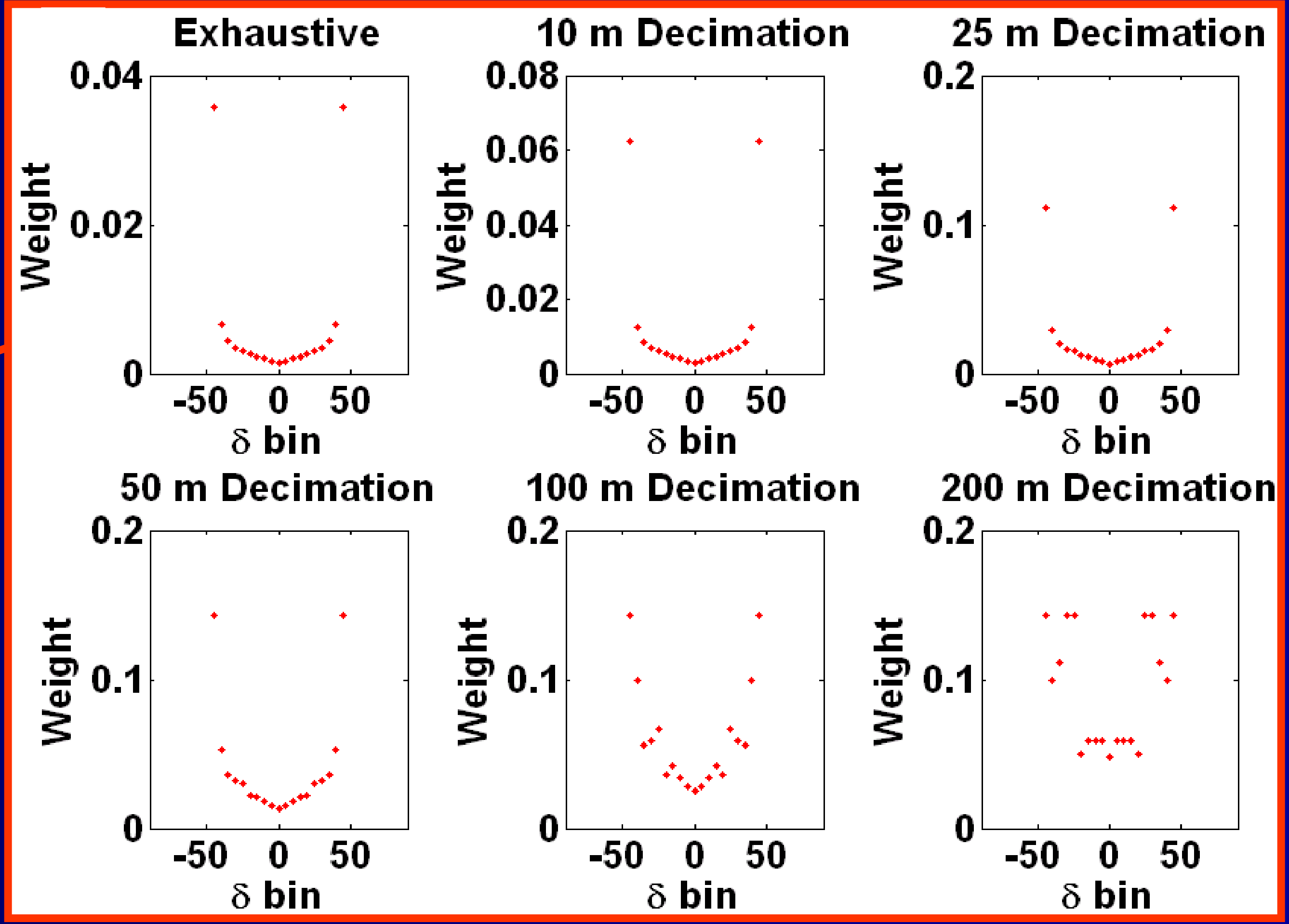




Method

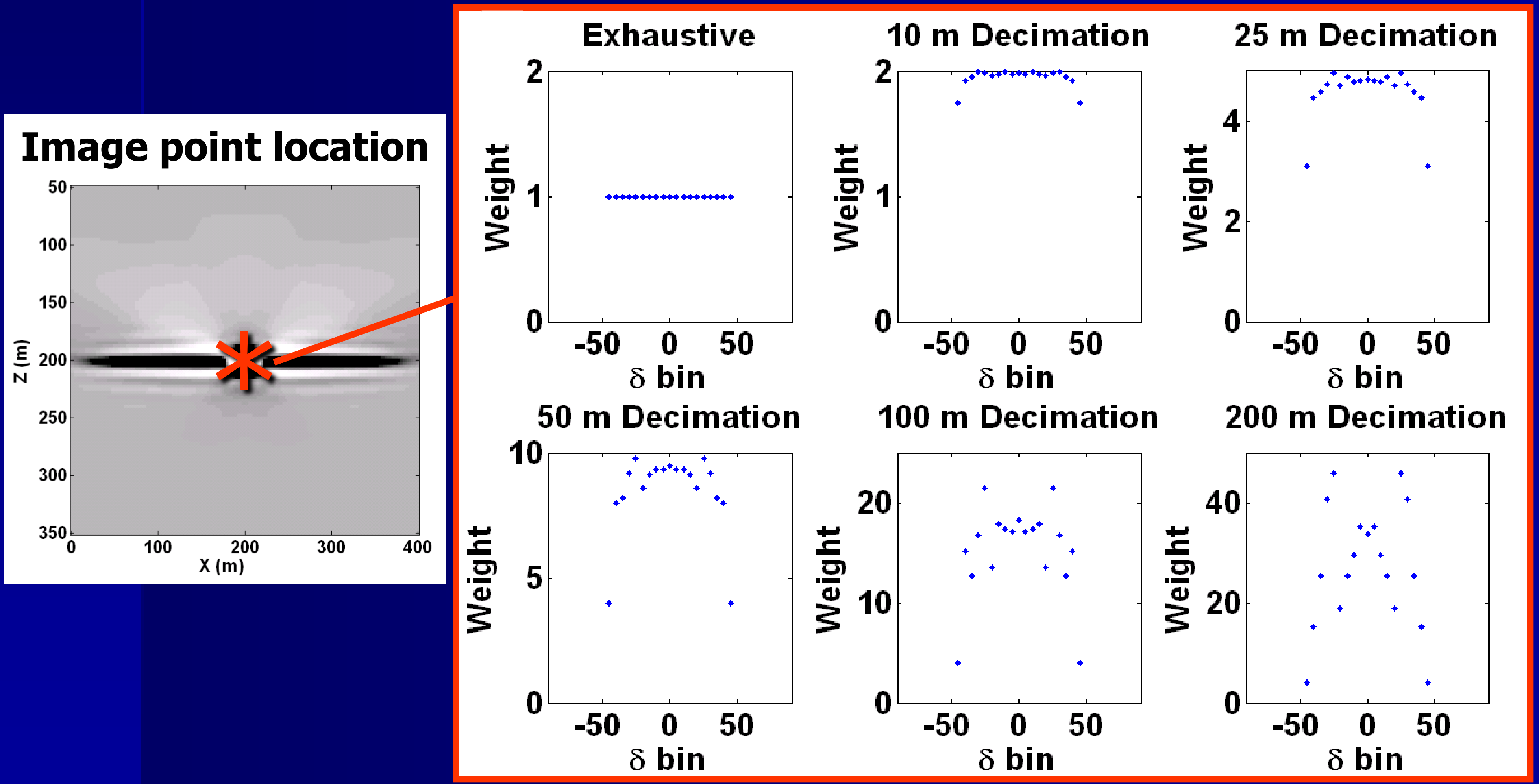
- Fold weights: 1/decimated_hits





Method

- Ratio weights: exh_hits/dec_hits



Method

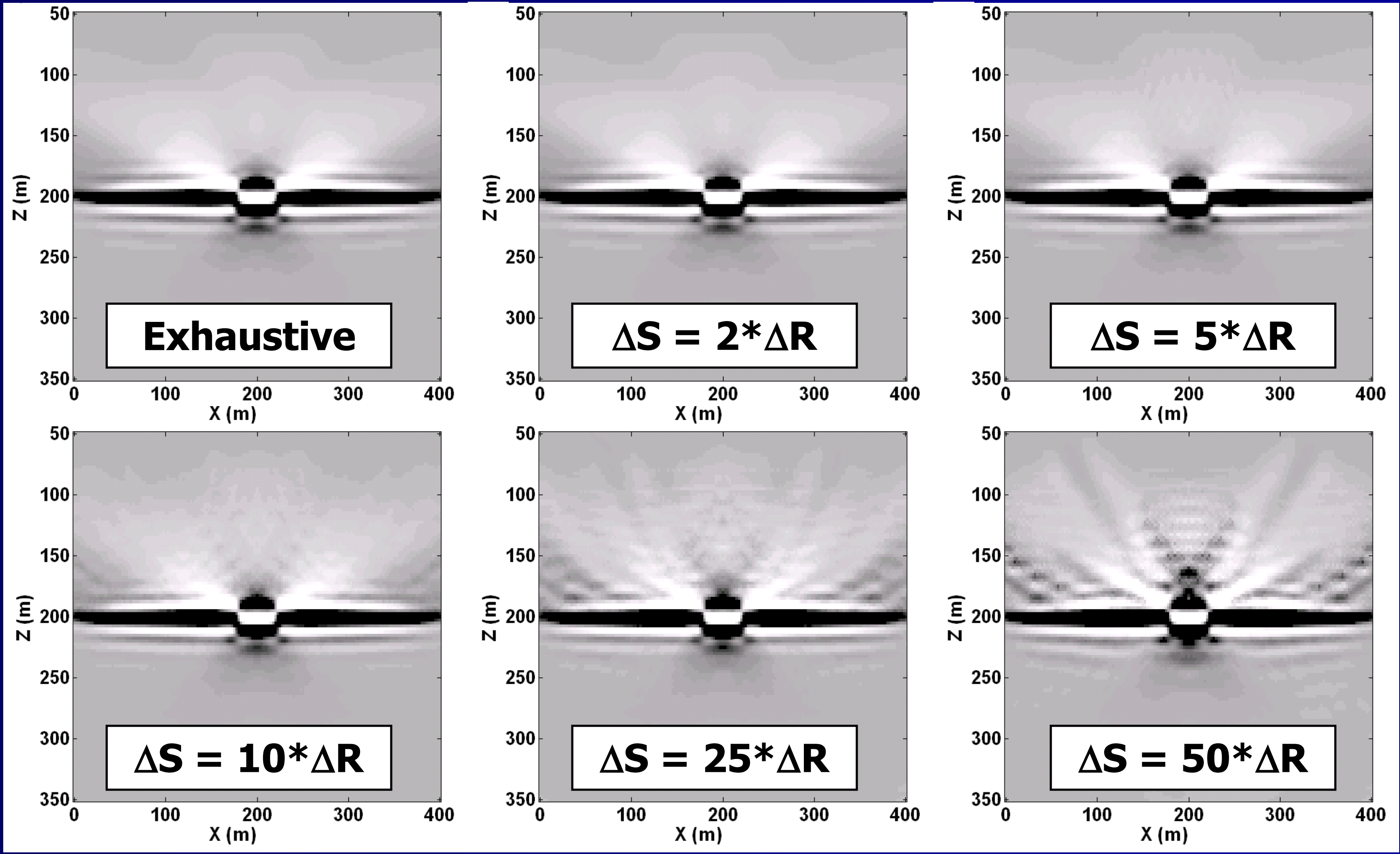
- Migrate each shot record into delta-limited volumes and apply weights during stacking:

$$\mathrm{Im}\left(x_i, y_i, z_i\right) = \sum_{j\,shots} \left[\sum_{k\,bins} W_k * \psi_j\left(x_i, y_i, z_i, \delta_k\right) \right]$$

- Or, precompute weights and apply during conventional migration, because weights are only a function of image point position and delta

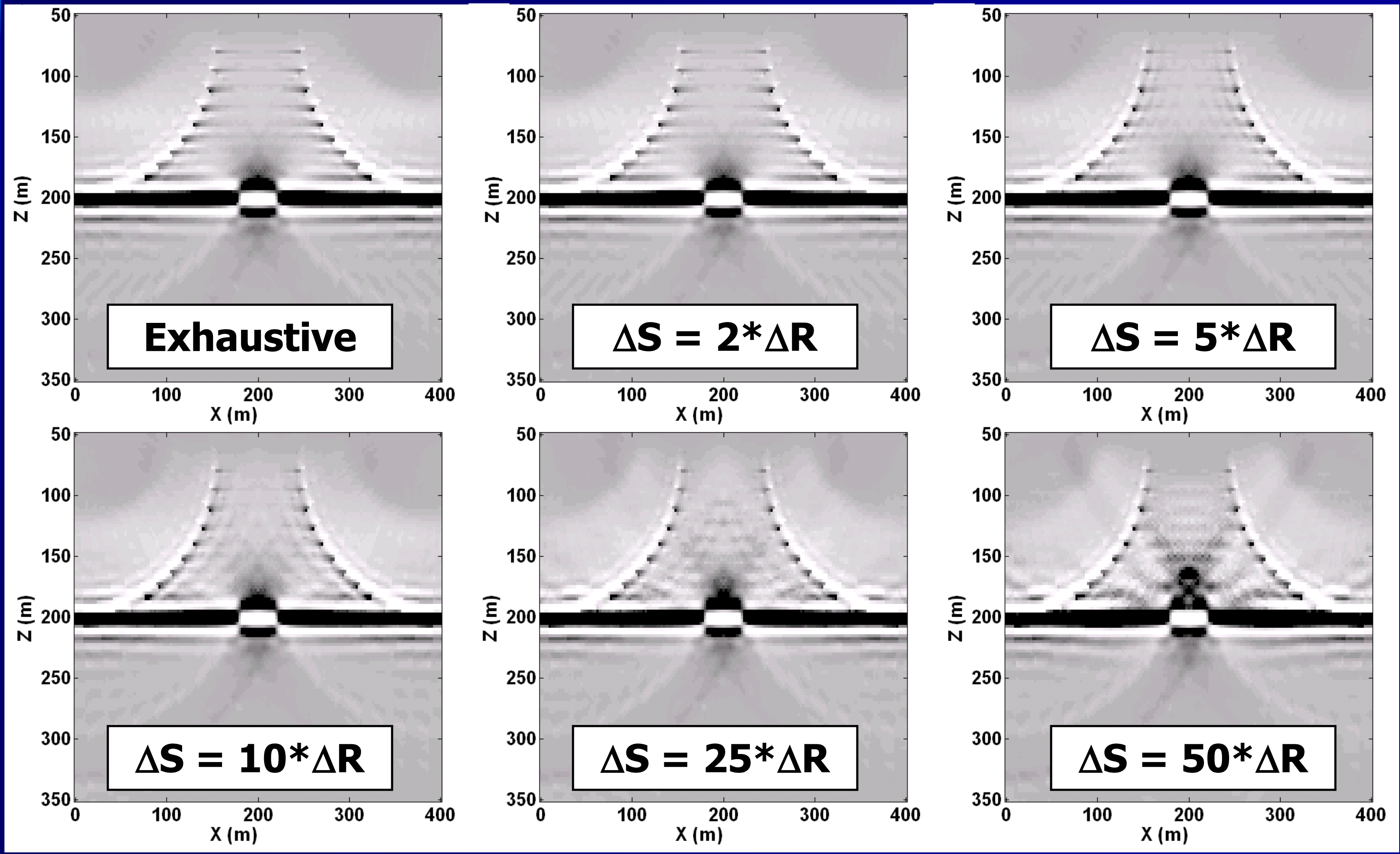
2D Application

- Results: delta ratio weights



2D Application

- Results: delta fold weights

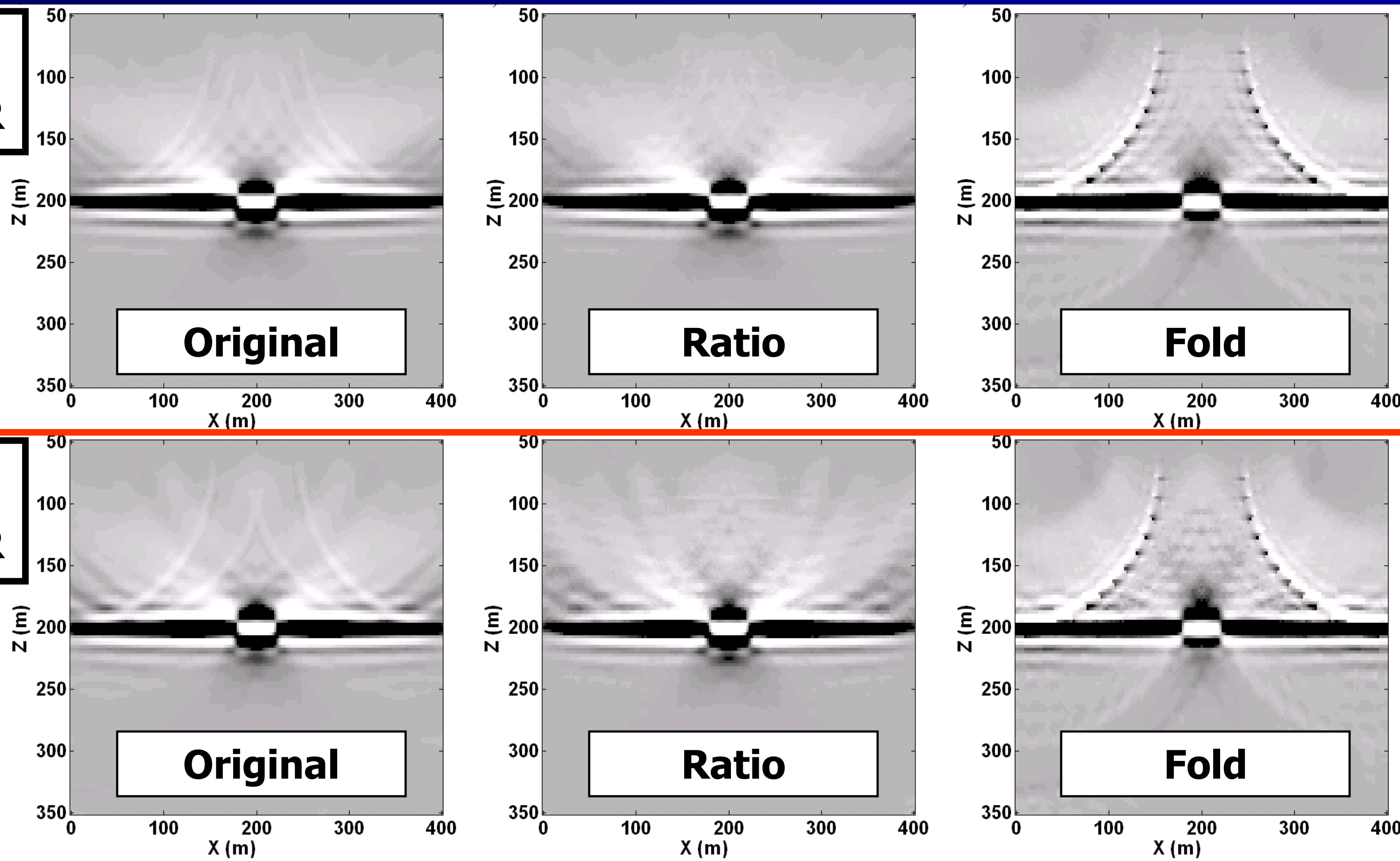


2D Application

- Comparison: ratio vs. fold weights

$\Delta S = 10 * \Delta R$

$\Delta S = 25 * \Delta R$

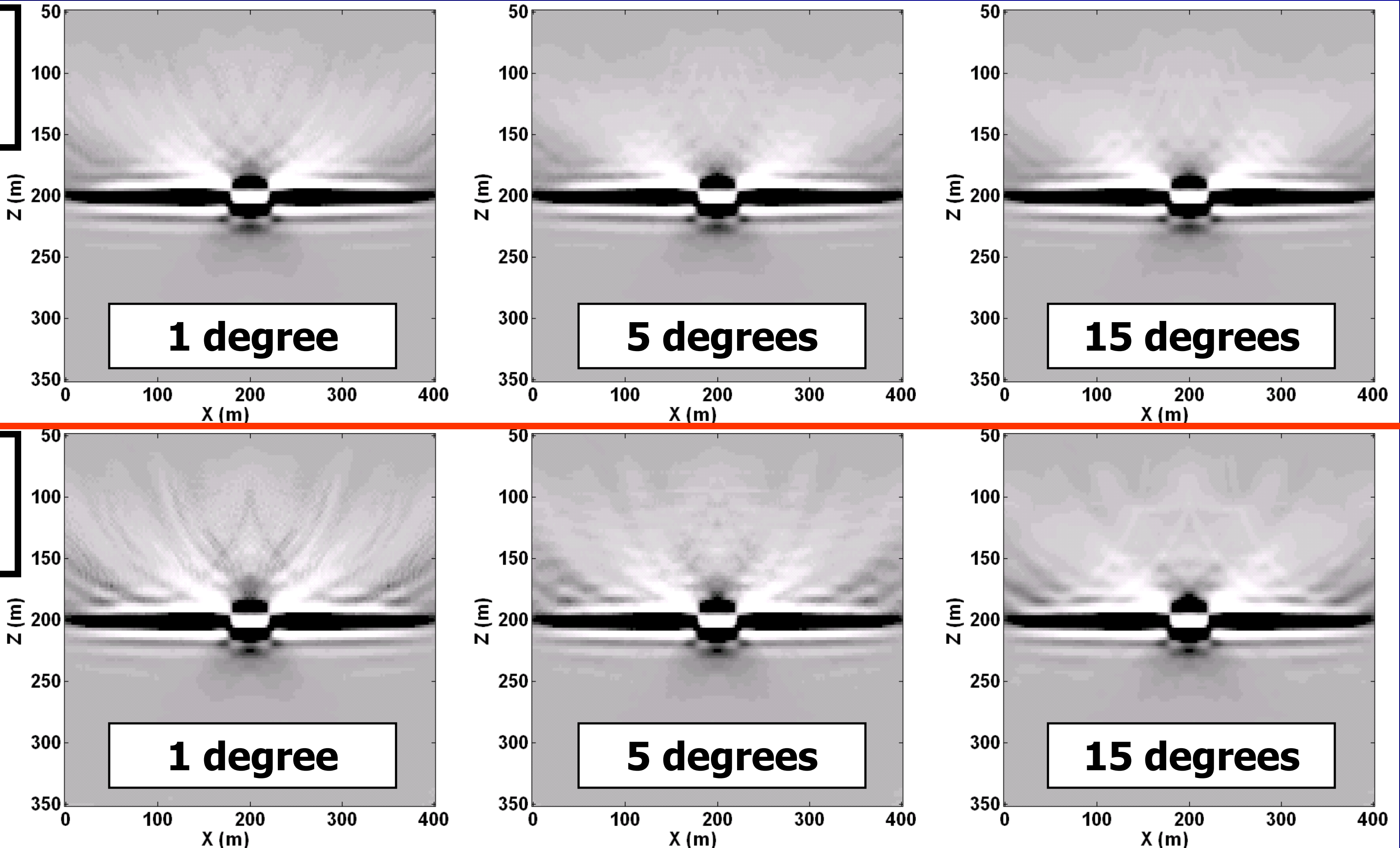


2D Application

- Comparison: bin widths (ratio weights)

ΔS = 10*ΔR

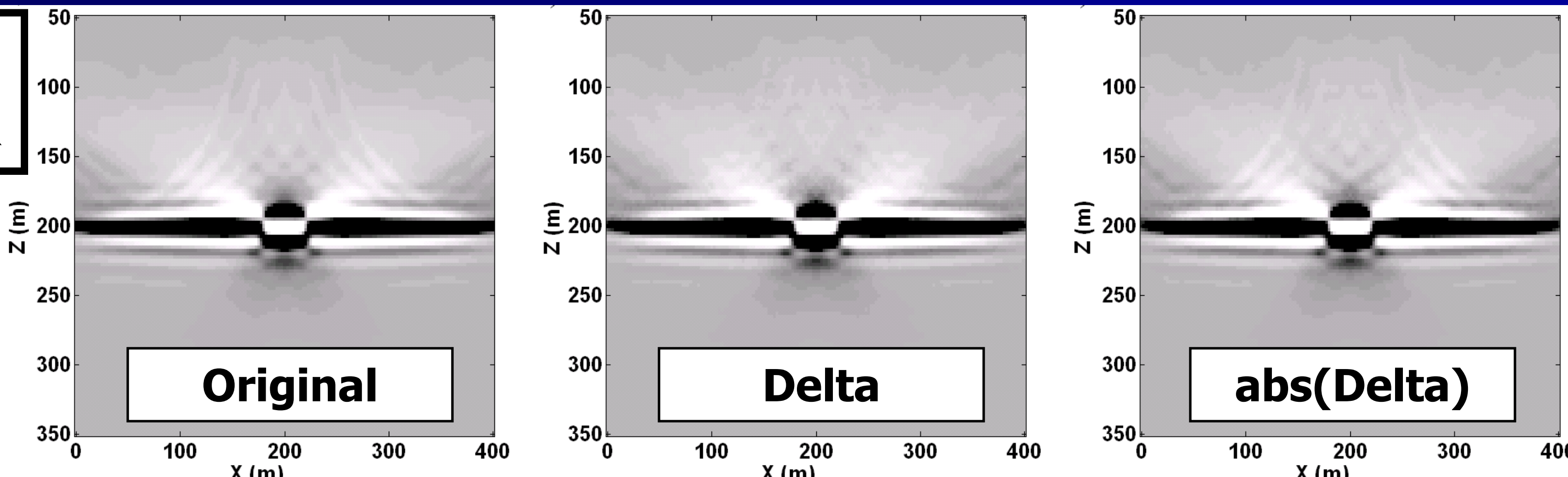
ΔS = 25*ΔR



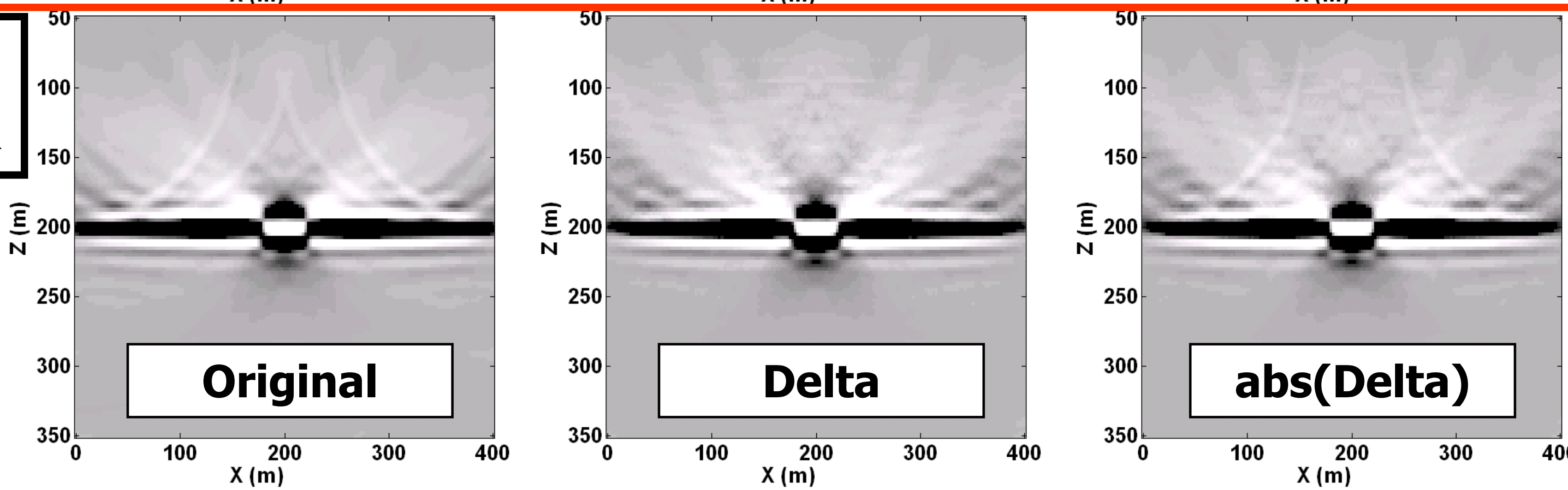
2D Application

- Comparison: delta ratio vs. abs(delta)

ΔS = **10***ΔR



ΔS = **25***ΔR



2D Observations

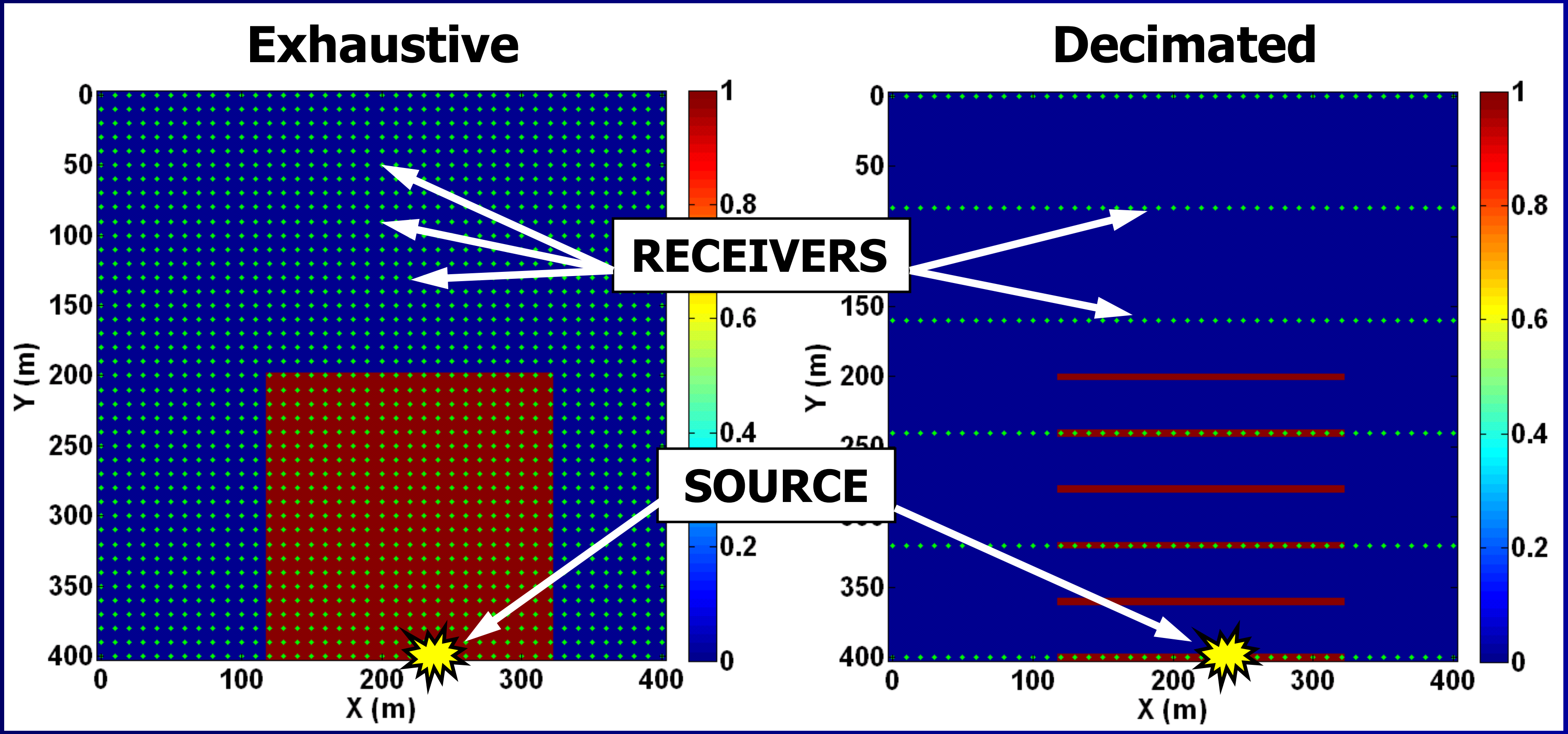
- Delta ratio weights appear to reduce footprint artefacts
- Delta fold weights compensate for aperture but enhance edge artefacts
- Bin width affects results
- Considering the sign of delta produces better results than abs(delta)

3D Method

- Full simulations, similar to in 2D are currently being produced
- Hit count maps for single shots show how the method will apply

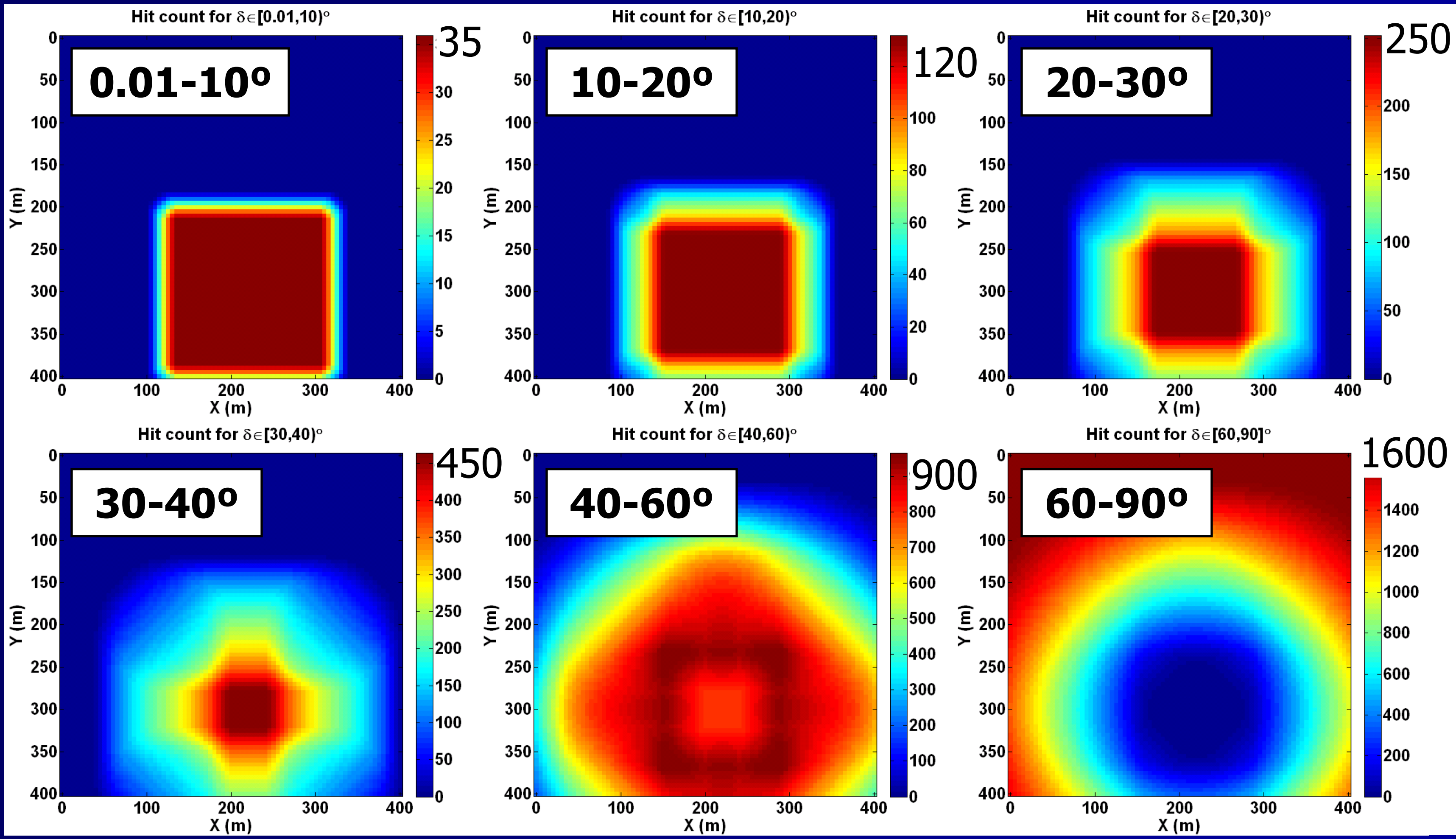
3D Delta Hit Counts

- Delta = 0° hit count is identical to CMP fold



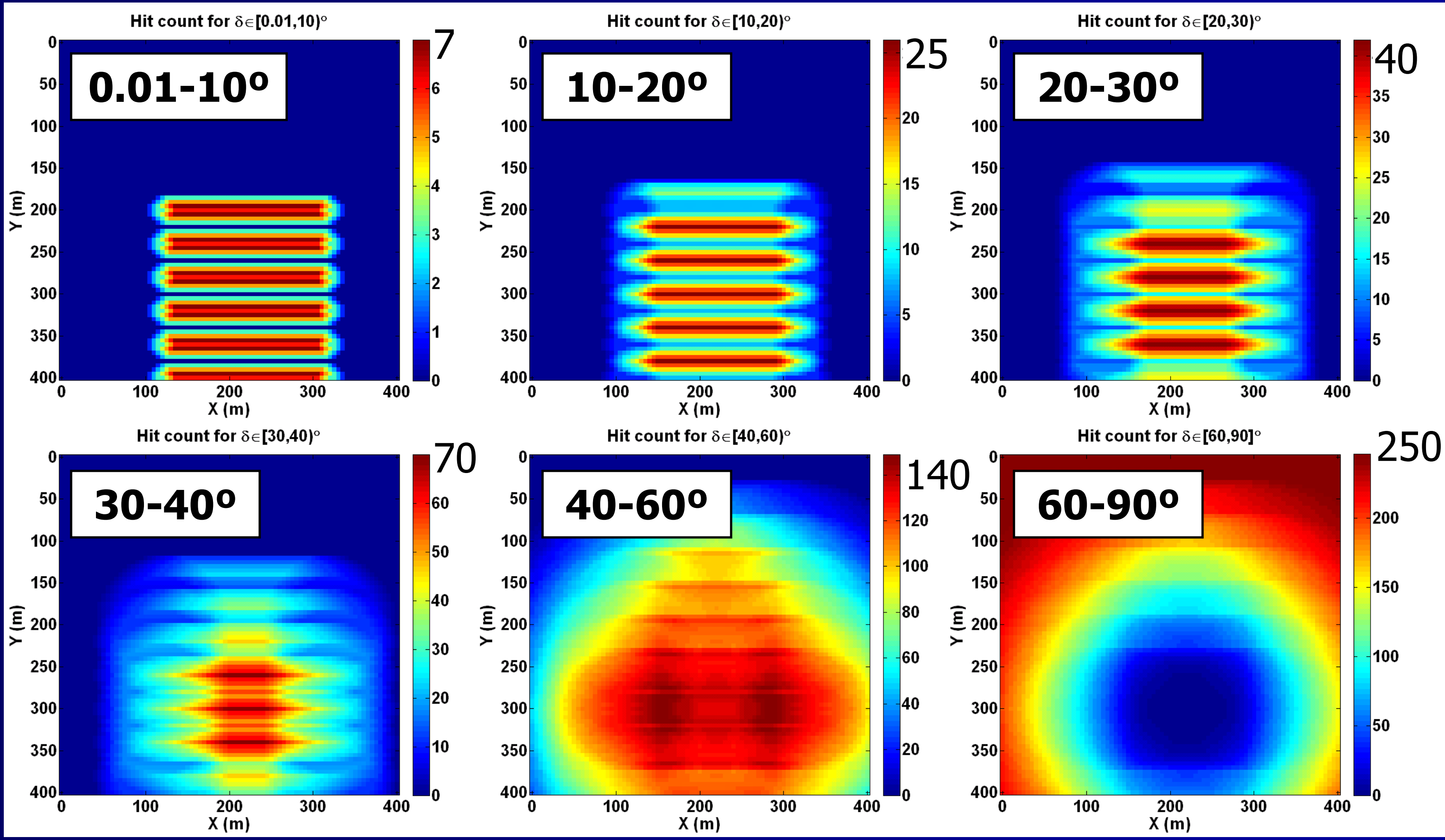
3D Delta Hit Counts

- Exhaustive survey non-zero deltas:



3D Delta Hit Counts

- Decimated survey non-zero deltas:



3D Observations

- Delta hit counts for single shots reflect differences in illumination between exhaustive and decimated datasets
- Delta weights result from summing the hit count maps for all shots

Conclusions

- Delta weights attempt to compensate for irregular image point illumination
- In 2D simulations, footprint appears to be reduced when delta ratio weights are applied during stacking of migrated shot records
- The method is similarly applicable in 3D

Future work

- More work determining optimal binning
- Implementation of Gaussian windowing
- Production of weighted stacks in 3D
- More work on theoretical weights

Acknowledgements

- Sponsors of CREWES
- Sponsors of POTSI
- NSERC
- Alberta Ingenuity