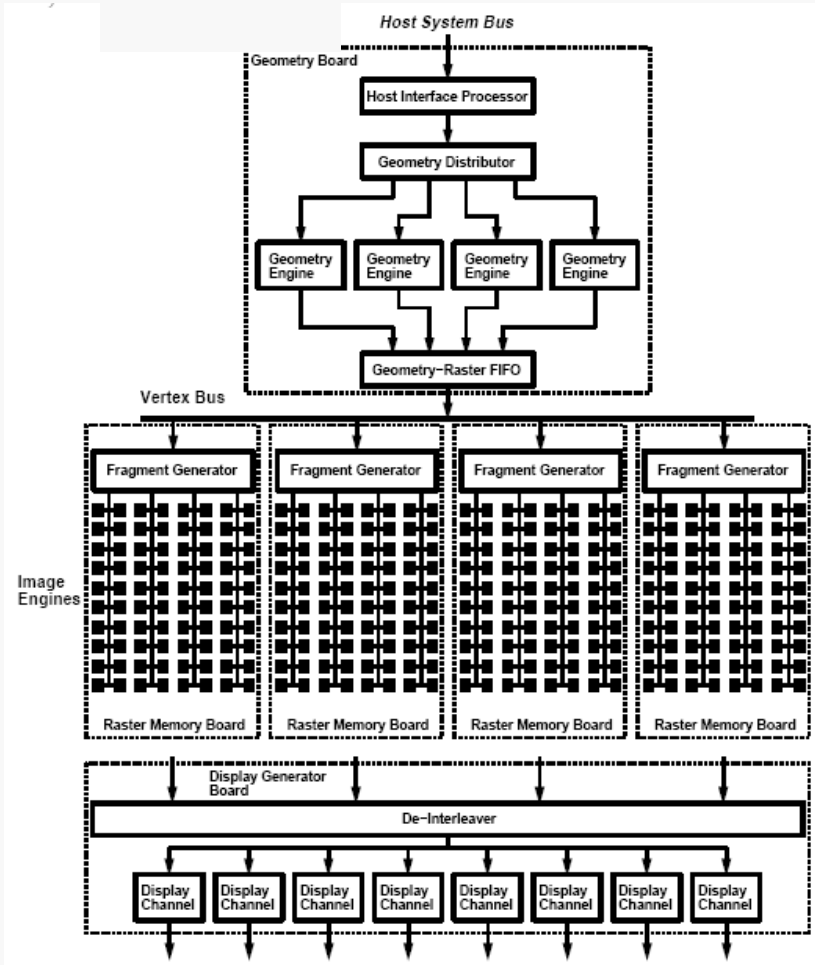
Microsoft® Research Asia

Minimum Triangle Separation for Correct Z-Buffer Occlusion

Kurt AKELEY and Jonathan SU
Microsoft Research Asia
3 September 2006

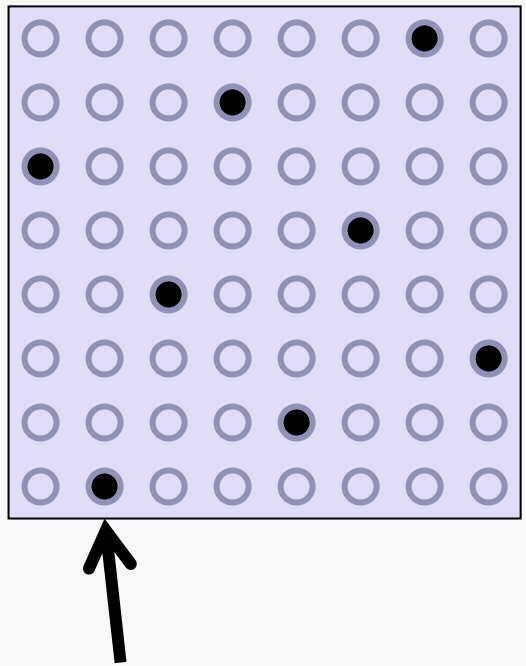
InfiniteReality



Figures taken from *InfiniteReality: A Real-Time Graphics System*, Proceedings of SIGGRAPH 1997

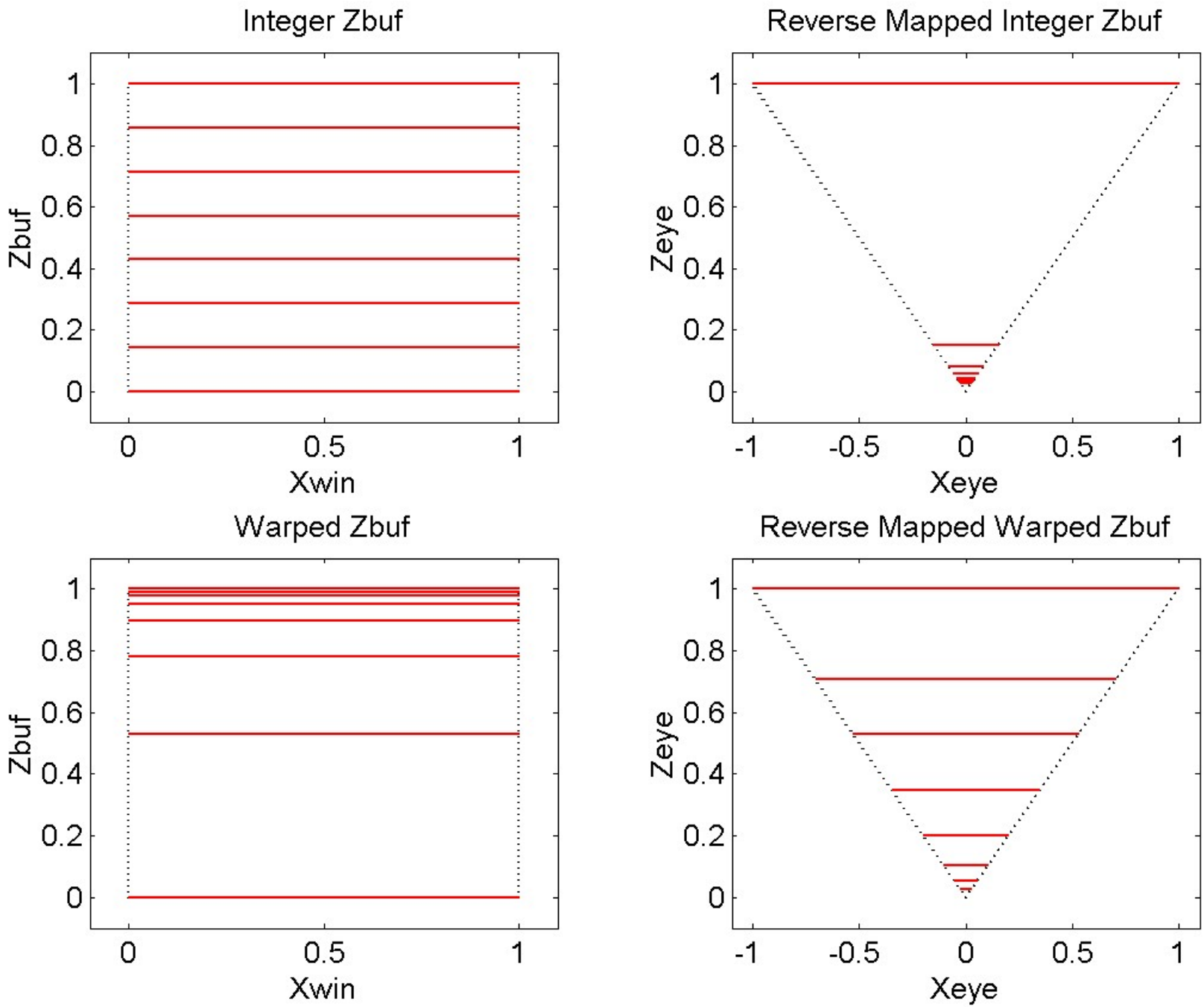
Multisample Antialiasing

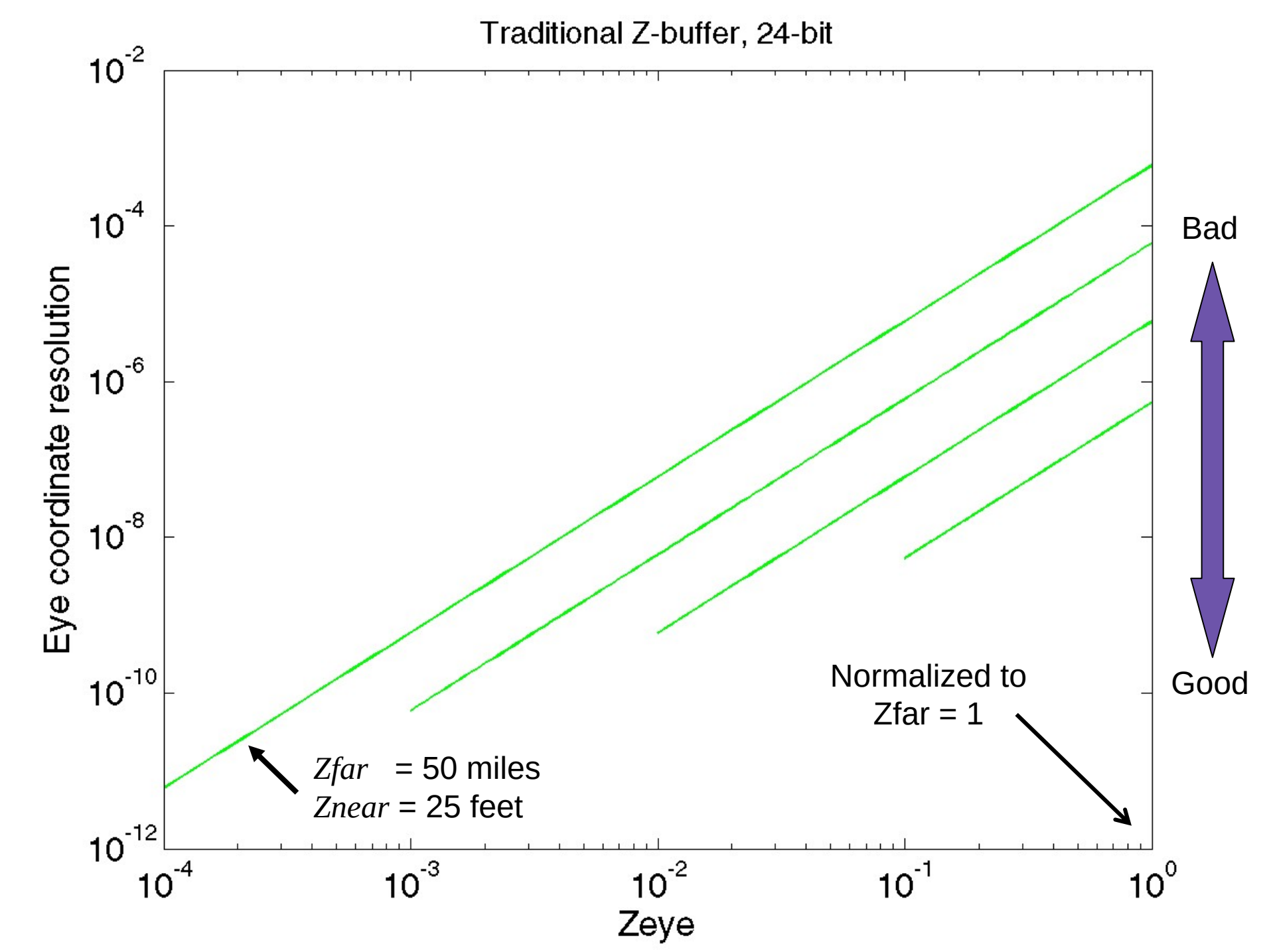
- Visual simulation requires high-quality and reliable antialiasing
- Eight samples per pixel
- Colors constant within fragments
  - Common cases compress well
  - Bandwidth reduction possible
- Z values vary within fragments
  - Required for interpenetration AA
  - Spatial compression is difficult
  - Bandwidth is excessive

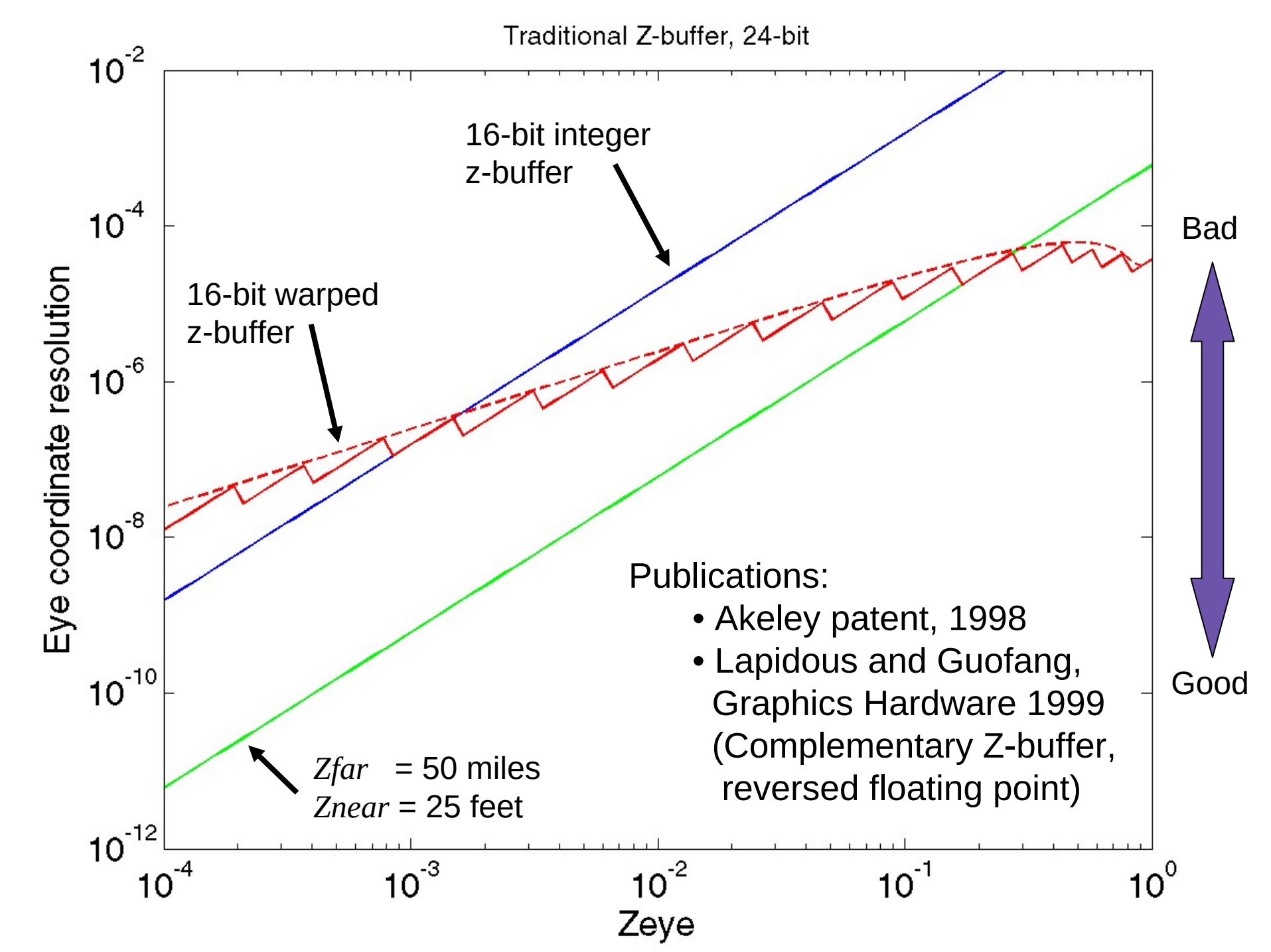


```
Struct sample {
    int_12 red;
    int_12 green;
    int_12 blue;
    int_24 z;
};
```

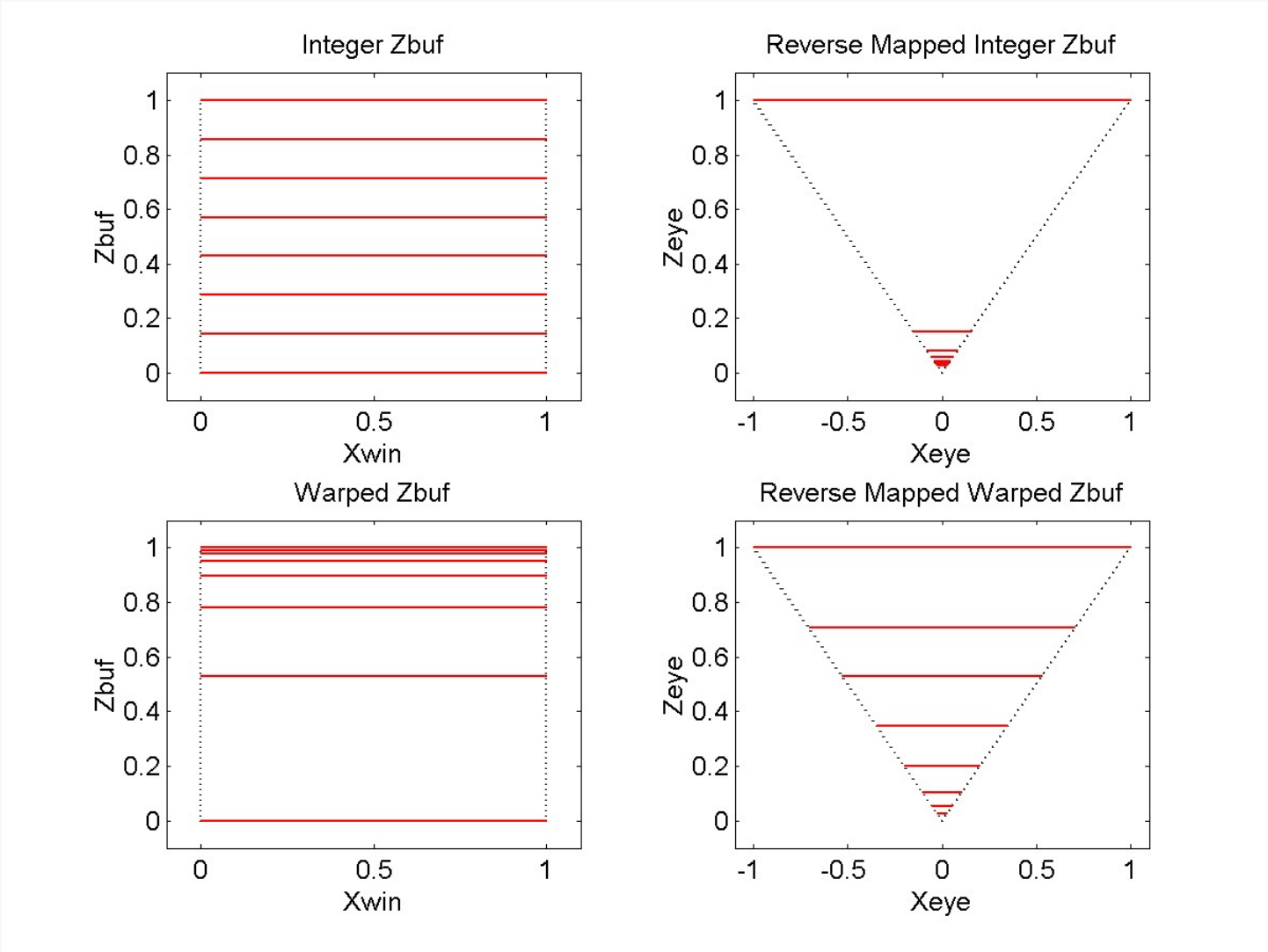
Can a 16-bit z value be used?


Integer Zbuf
Zbuf
Xwin
0
0.2
0.4
0.6
0.8
1
0
0.5
1
Reverse Mapped Integer Zbuf
Zeye
Xeye
-1
-0.5
0
0.5
1
Warped Zbuf
Reverse Mapped Warped Zbuf

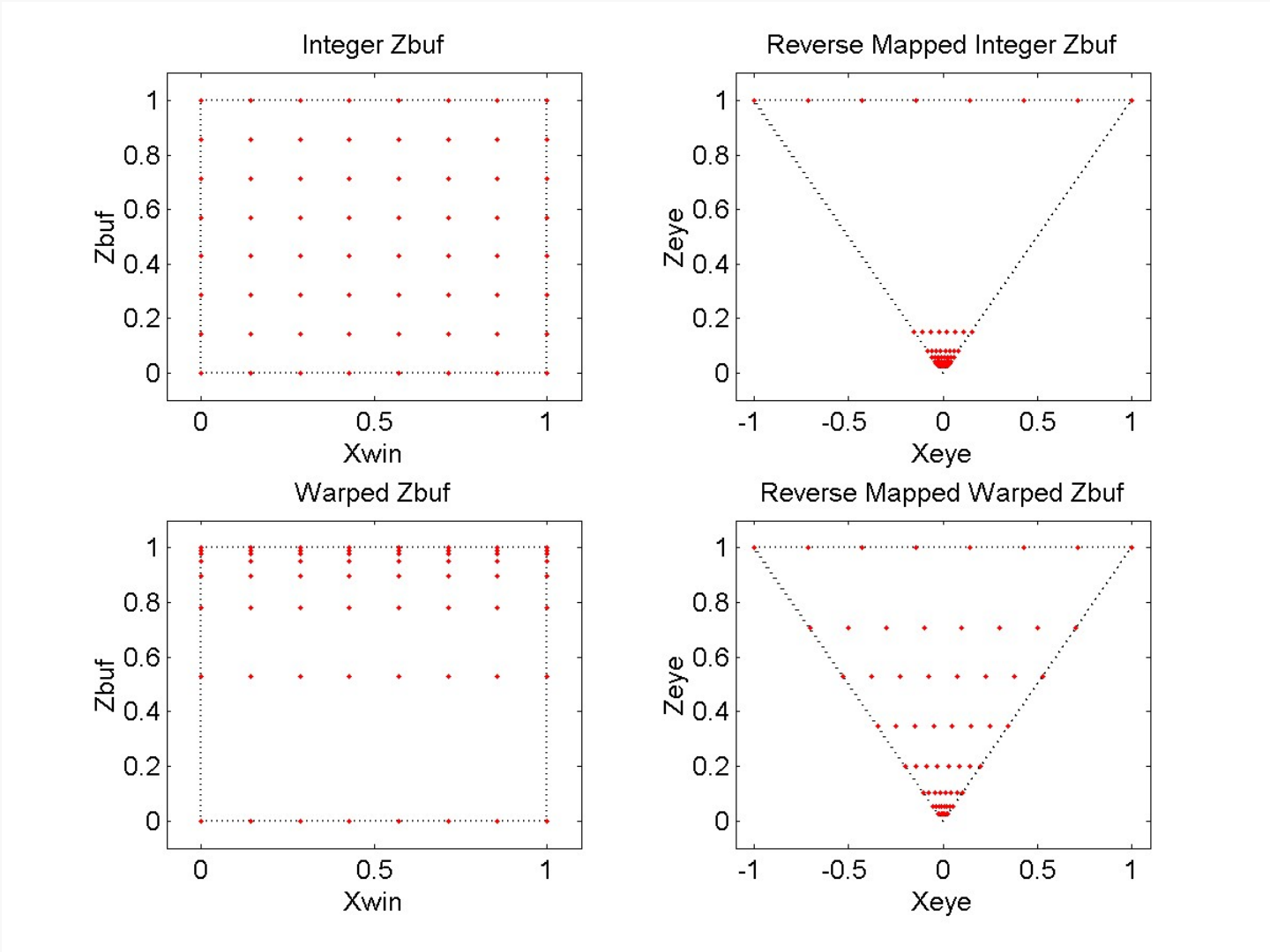

Traditional Z-buffer, 24-bit
Eye coordinate resolution
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-4}
10^{-3}
10^{-2}
10^{-1}
10^{0}
Zeye
Bad
Good
Normalized to
Zfar = 1
Zfar = 50 miles
Znear = 25 feet


Traditional Z-buffer, 24-bit
16-bit integer
z-buffer
16-bit warped
z-buffer
Zfar = 50 miles
Znear = 25 feet
Publications:
• Akeley patent, 1998
• Lapidous and Guofang,
Graphics Hardware 1999
(Complementary Z-buffer,
reversed floating point)
Bad
Good
Eye coordinate resolution
Zeye
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-4
10^-3
10^-2
10^-1
10^0

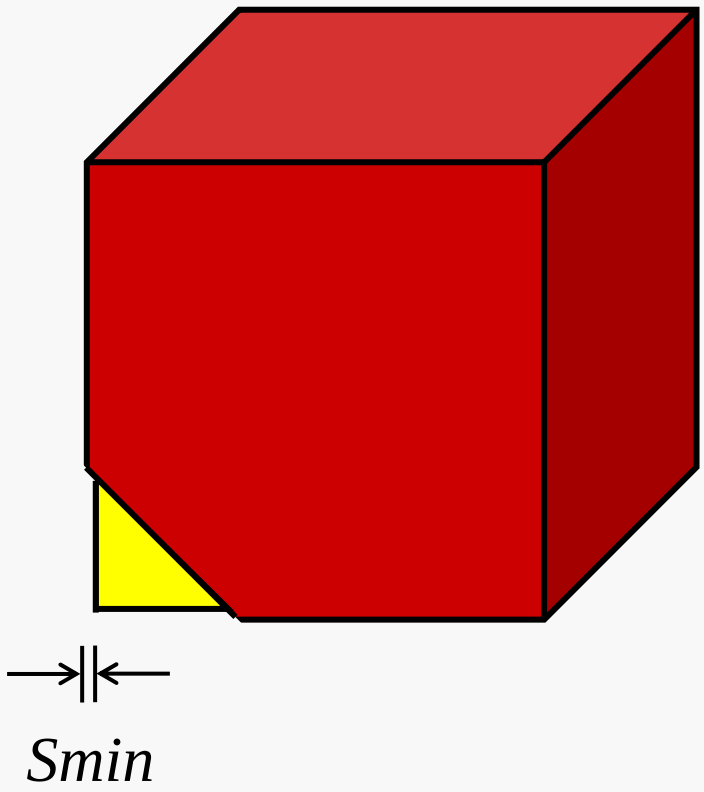
Ignores *Xwin* and *Ywin* Representation



Discrete *Xwin* and *Ywin* affect *Zbuf* values



Minimum Separation
(Z-buffer resolution is like vacuum cleaner horsepower)



Outline of Talk

- List assumptions and definitions
- Compute *Smin*
  - Treating *Xwin*, *Ywin*, and *Zbuf* as discrete representations
  - Treating all other arithmetic as ideal
- Compute *Smin* again
  - Accumulating error due to all discrete representations from eye-coordinates to the z-buffer
    - Ignoring clipping
    - Assuming high-quality rasterization
- Suggest guidance opportunities
- Conclude

We verify all analysis with simulation results

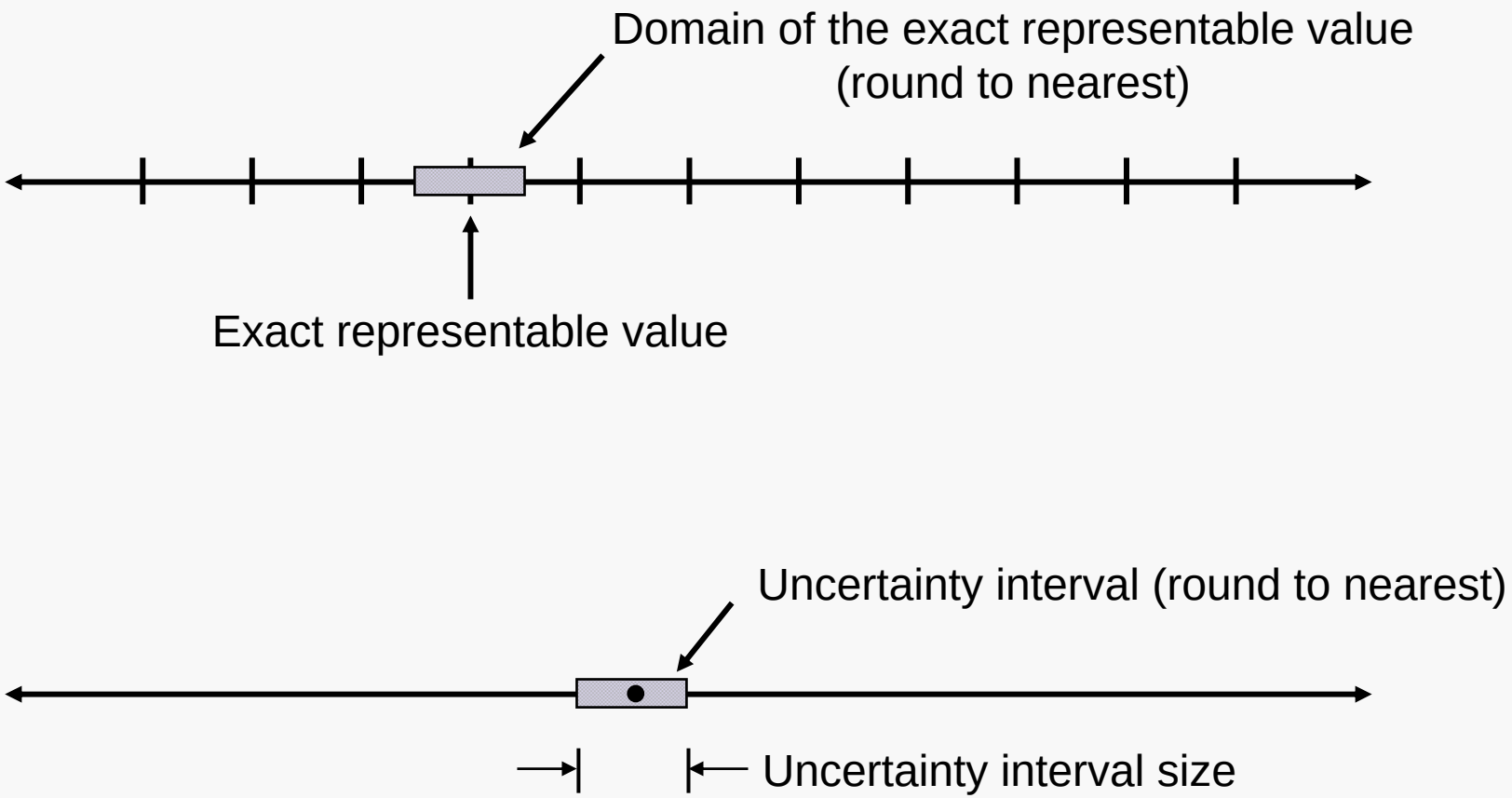
Assumptions and Definitions

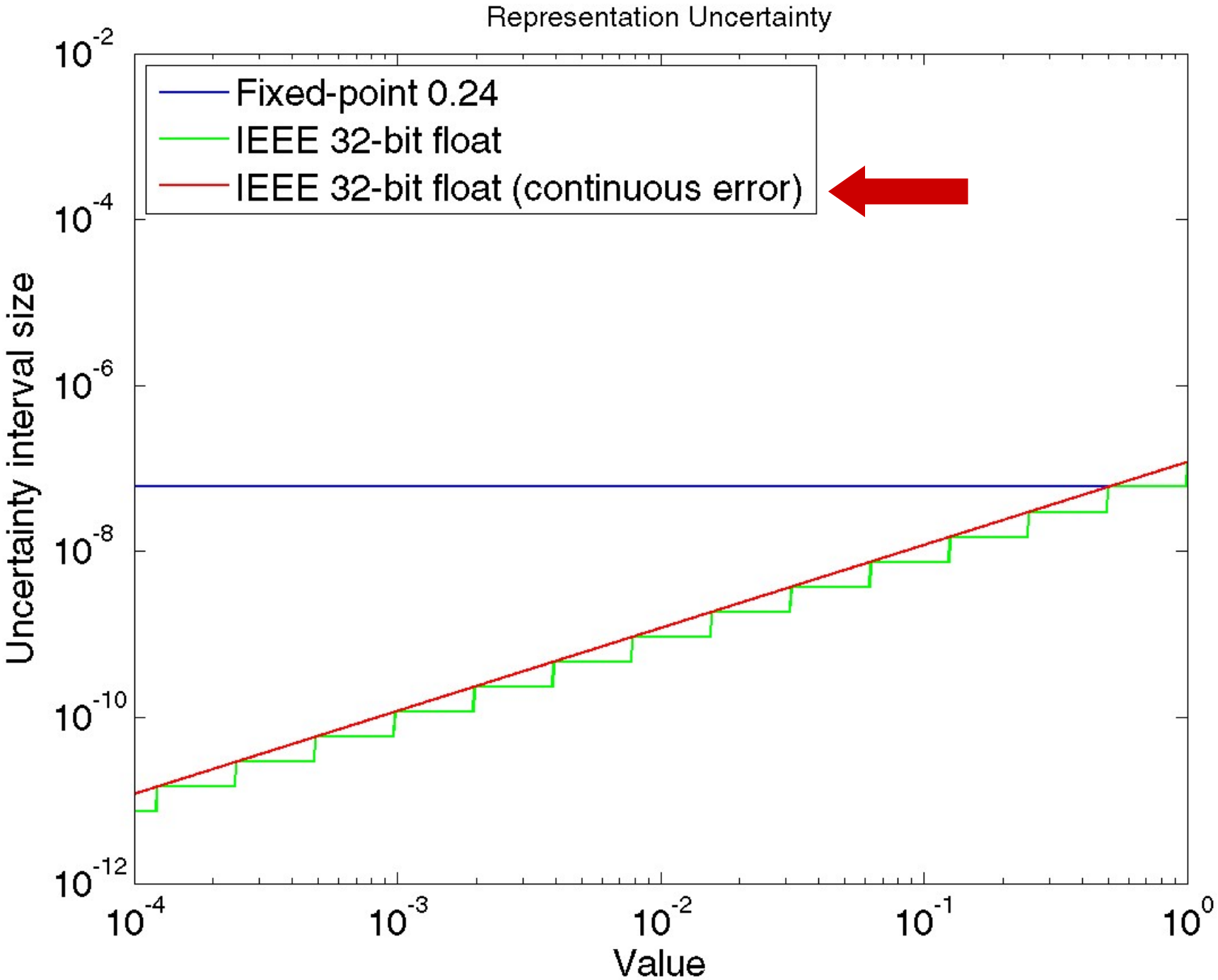
Assumptions

- Standard OpenGL/Direct3D-like pipeline
  - Used in standard way (e.g., not ray tracing)

- Triangles only
  - No points
  - No lines

Uncertainty Intervals



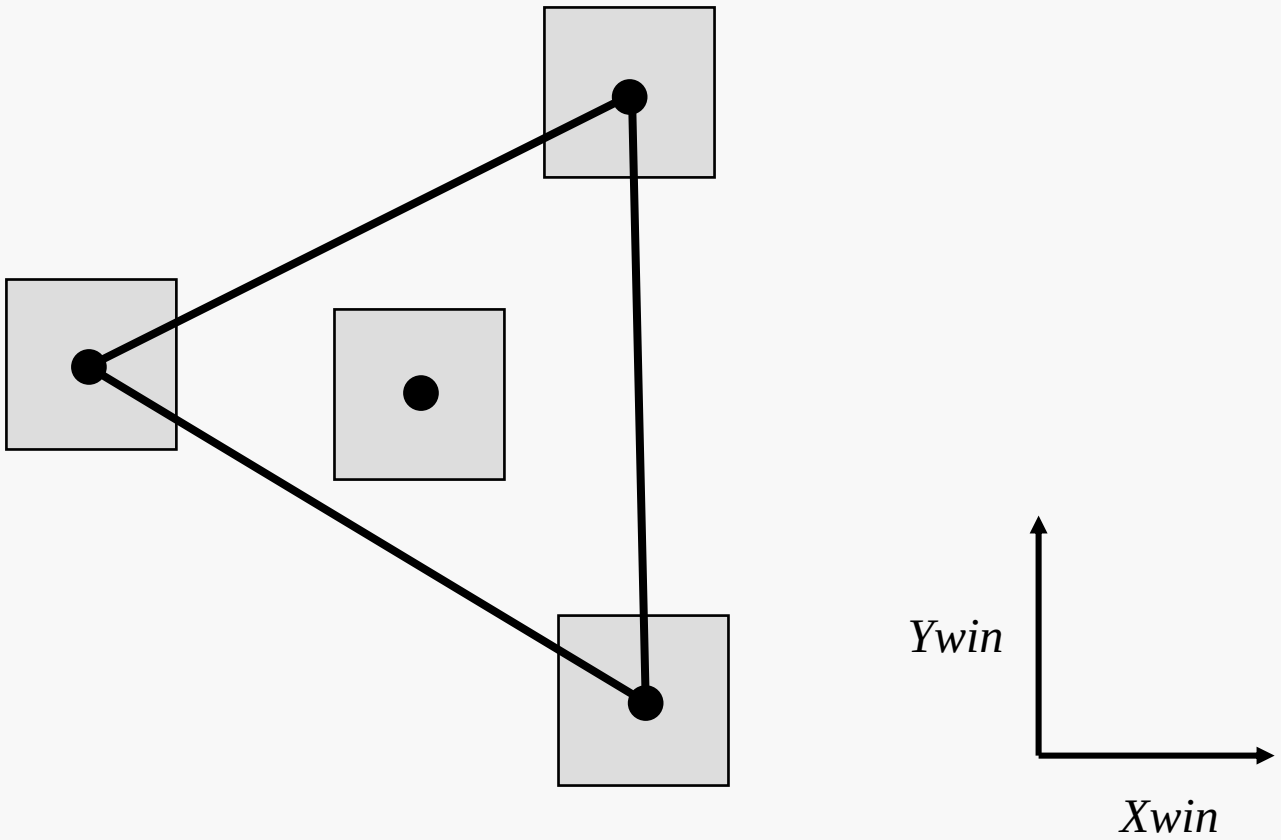

Representation Uncertainty
Fixed-point 0.24
IEEE 32-bit float
IEEE 32-bit float (continuous error)
Uncertainty interval size
Value
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-4
10^-3
10^-2
10^-1
10^0

Compute *Smin*

(with discrete *Xwin*, *Ywin*, and *Zbuf*)

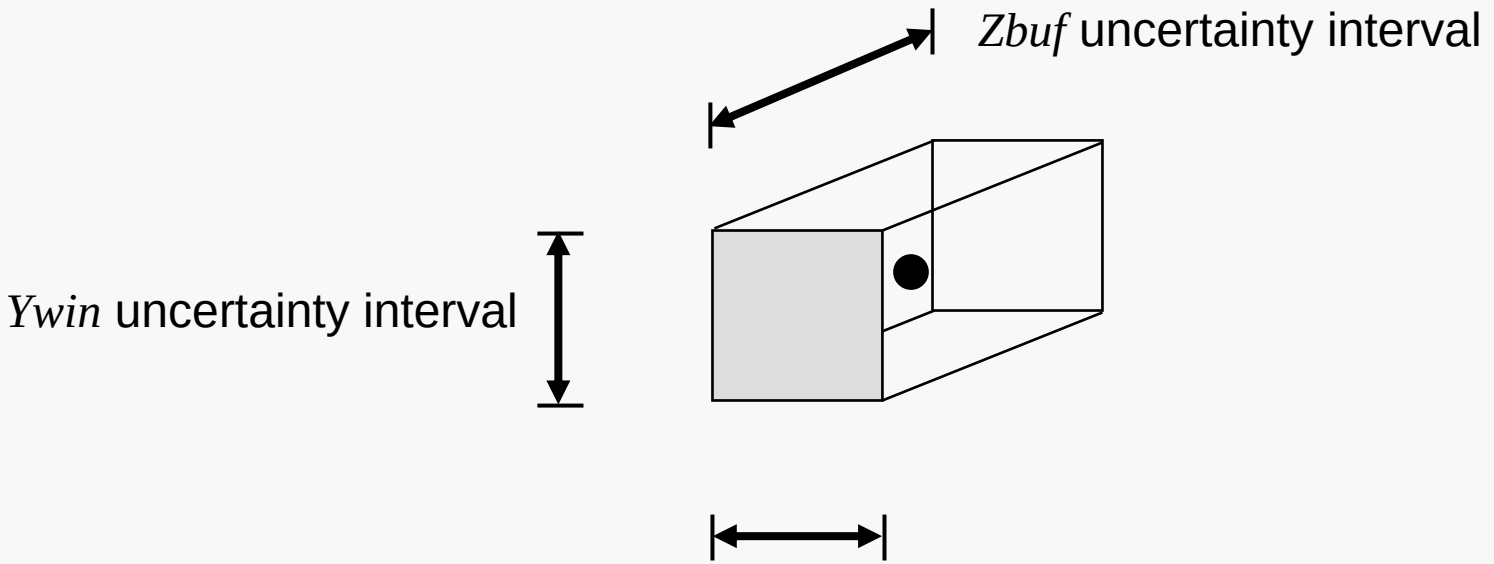
Xwin,Ywin Uncertainty Square

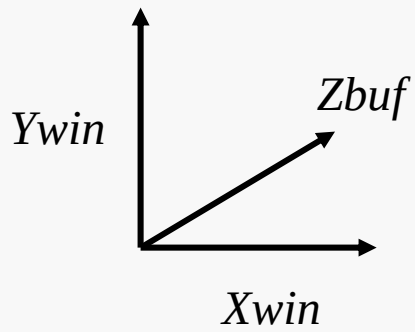
For any x,y location in the window-coordinate triangle



Uncertainty Cuboid

For any point on the surface of the 3-D triangle



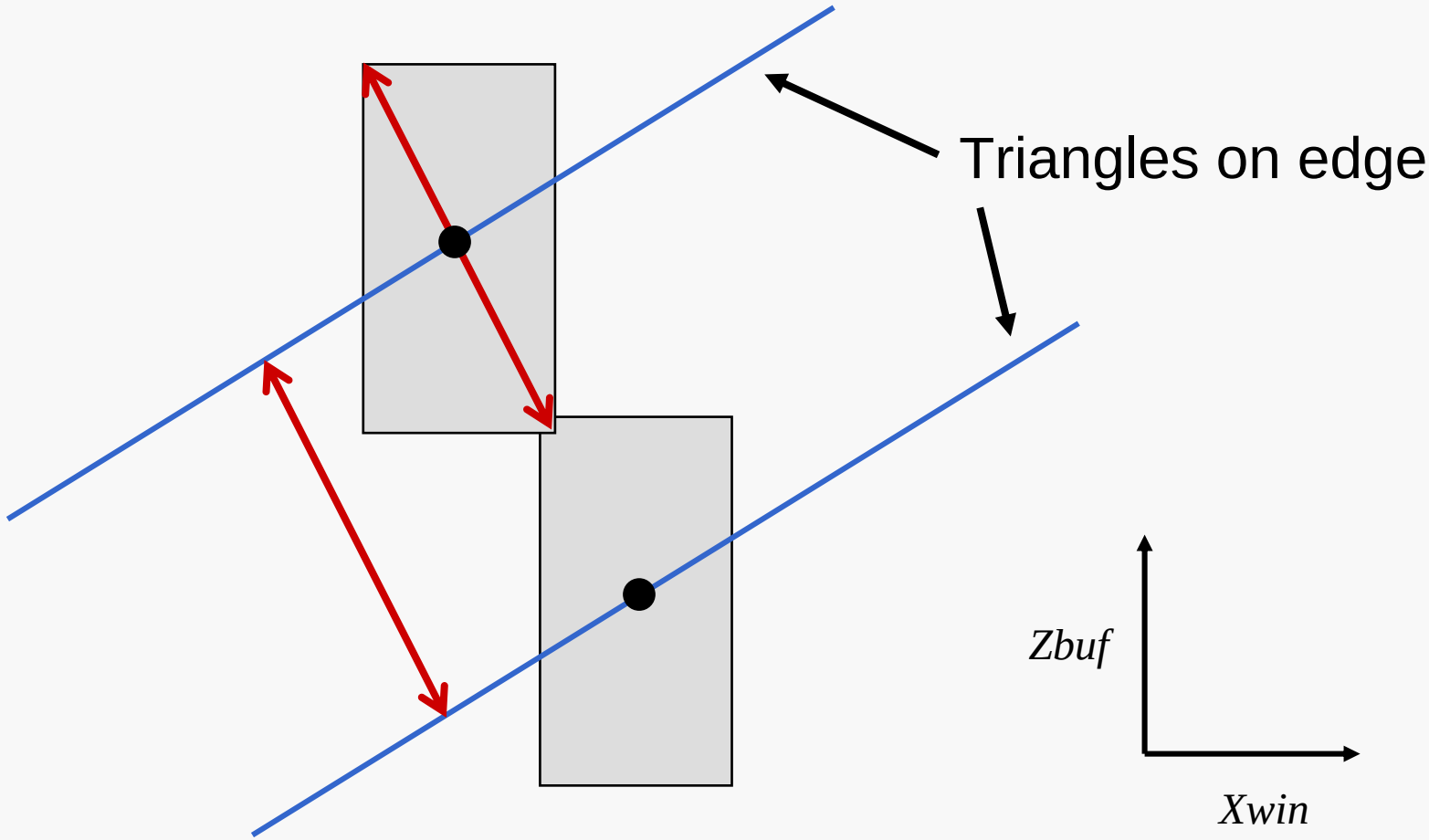


What is *Smin* ?

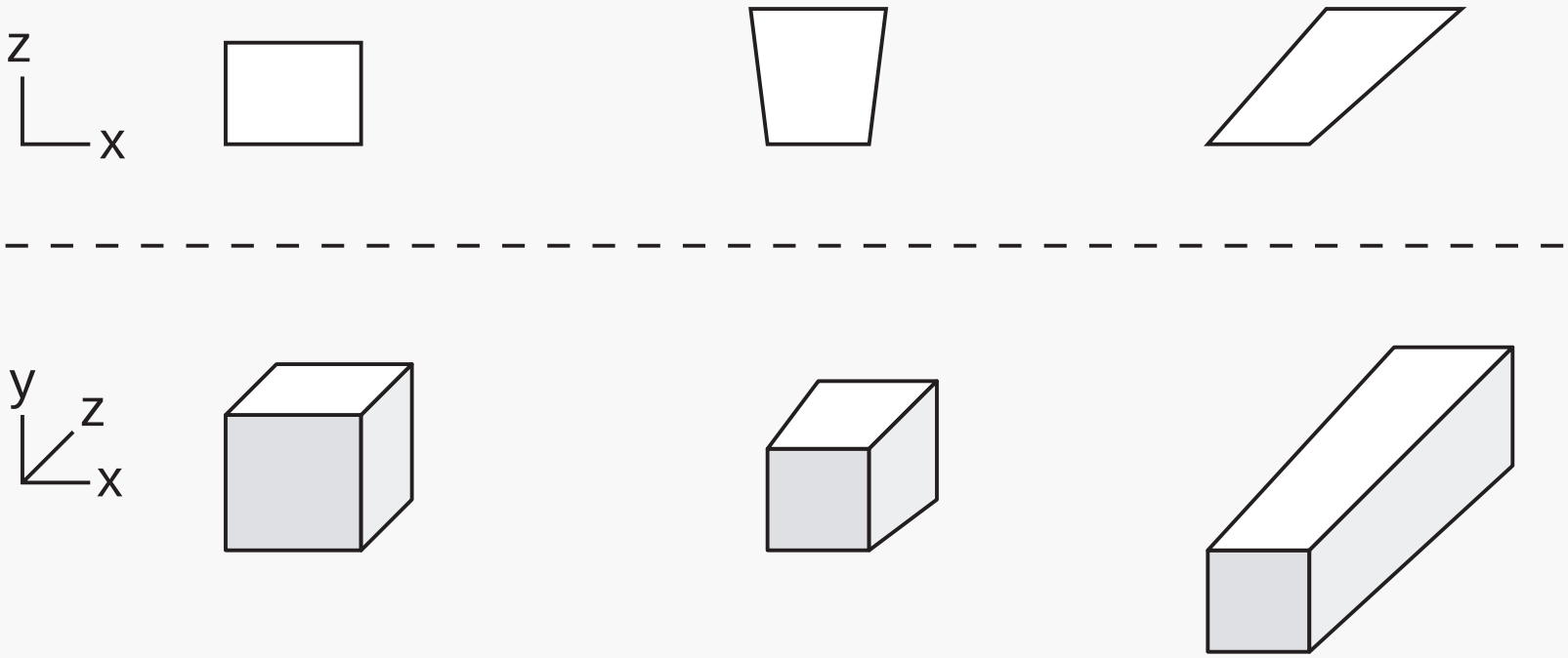
- Considering only *Zbuf*
  - Depth of the uncertainty cuboid
- Considering only *Xwin* and *Ywin*
  - Length of diagonal of the uncertainty square
- Considering *Xwin*, *Ywin*, and *Zbuf*
  - Length of diagonal of the uncertainty cuboid

Worst case is for triangles that are perpendicular to the diagonal

Uncertainty Volume Overlap → Failure



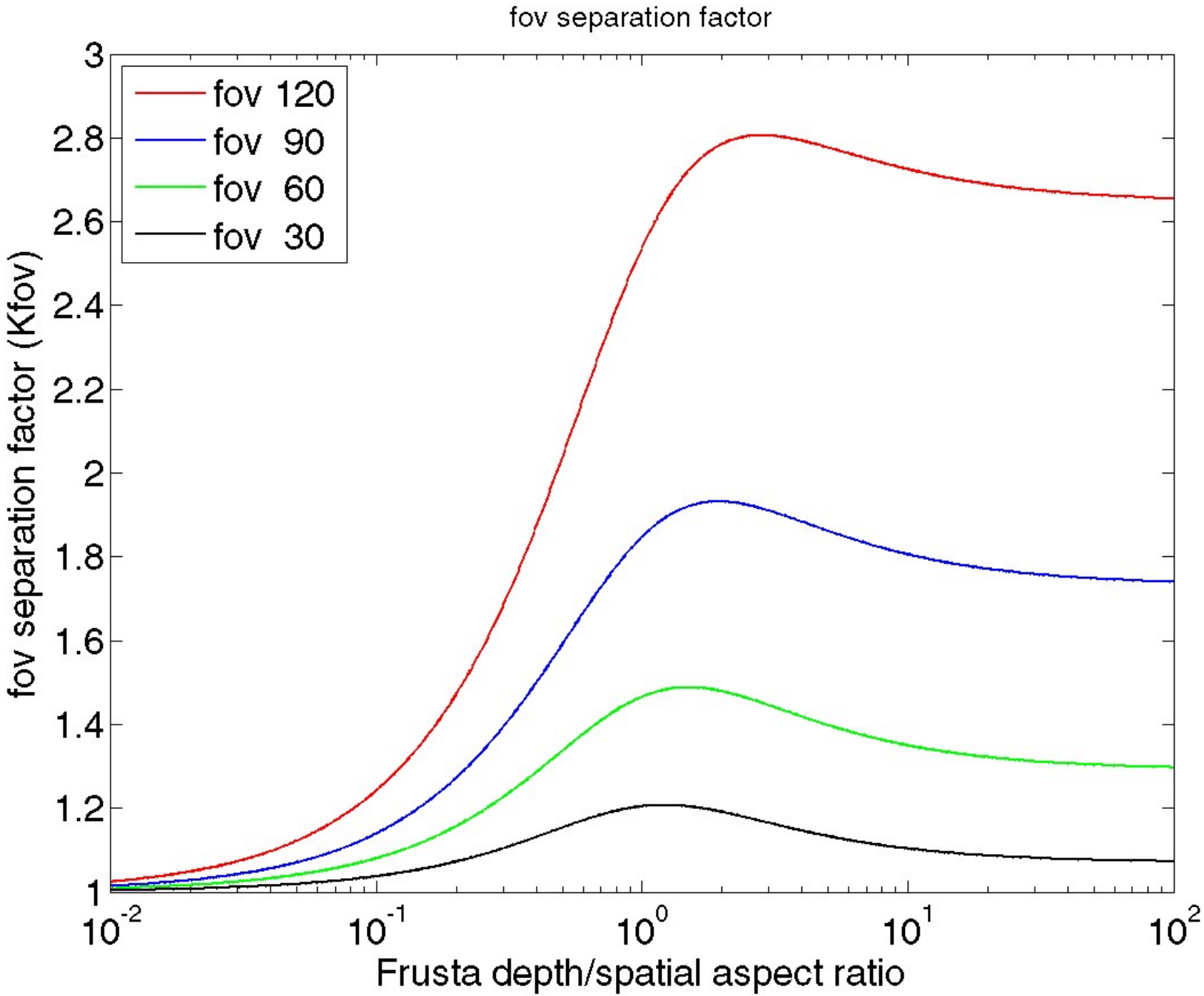
Wide Field of View Increases *Smin*

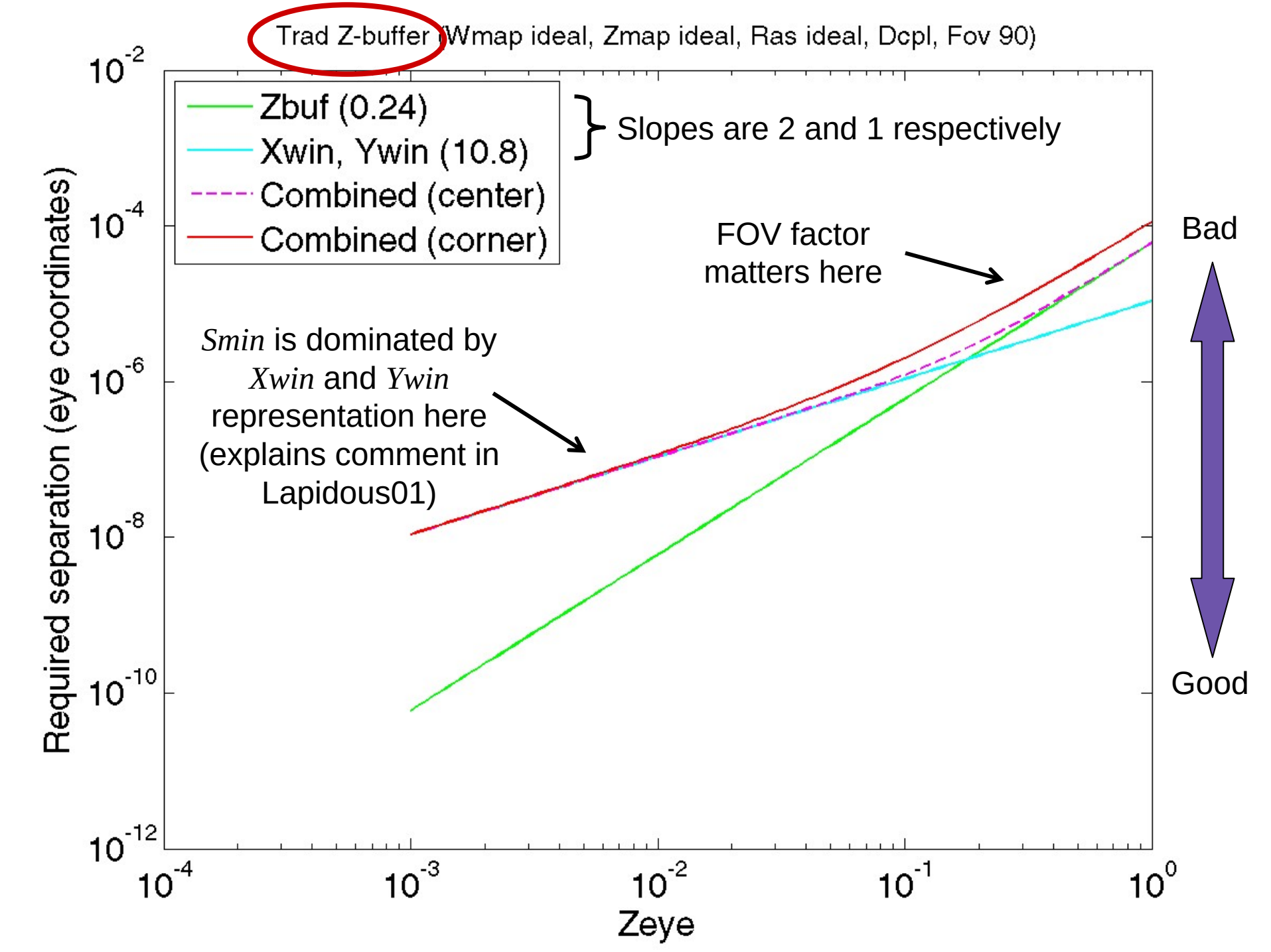


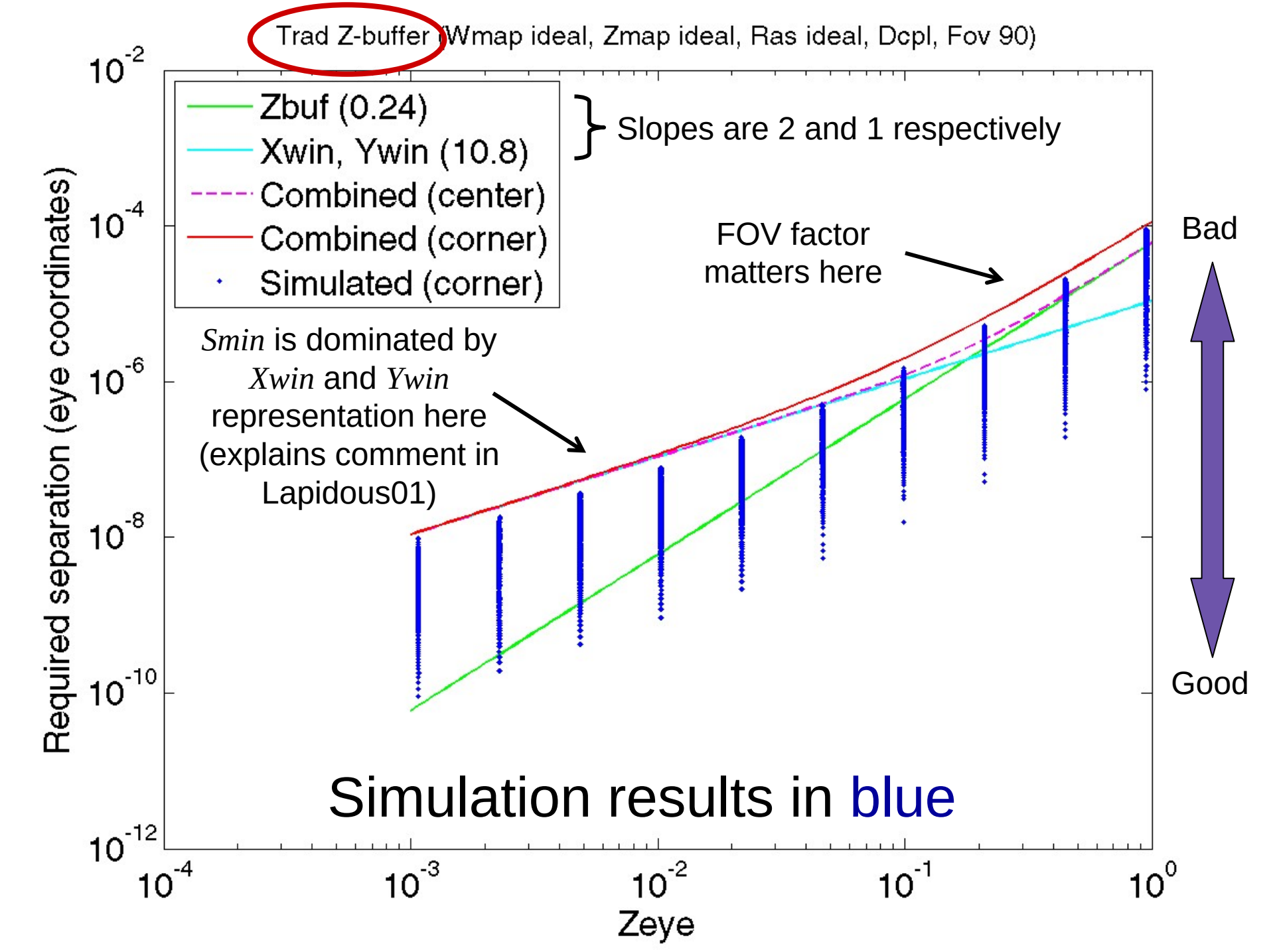
Window coordinates (a)

Eye coordinates (center screen) (b)

Eye coordinates (corner screen) (c)


fov separation factor
fov 120
fov 90
fov 60
fov 30
fov separation factor (Kfov)
Frusta depth/spatial aspect ratio
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
1.2
1
10^-2
10^-1
10^0
10^1
10^2

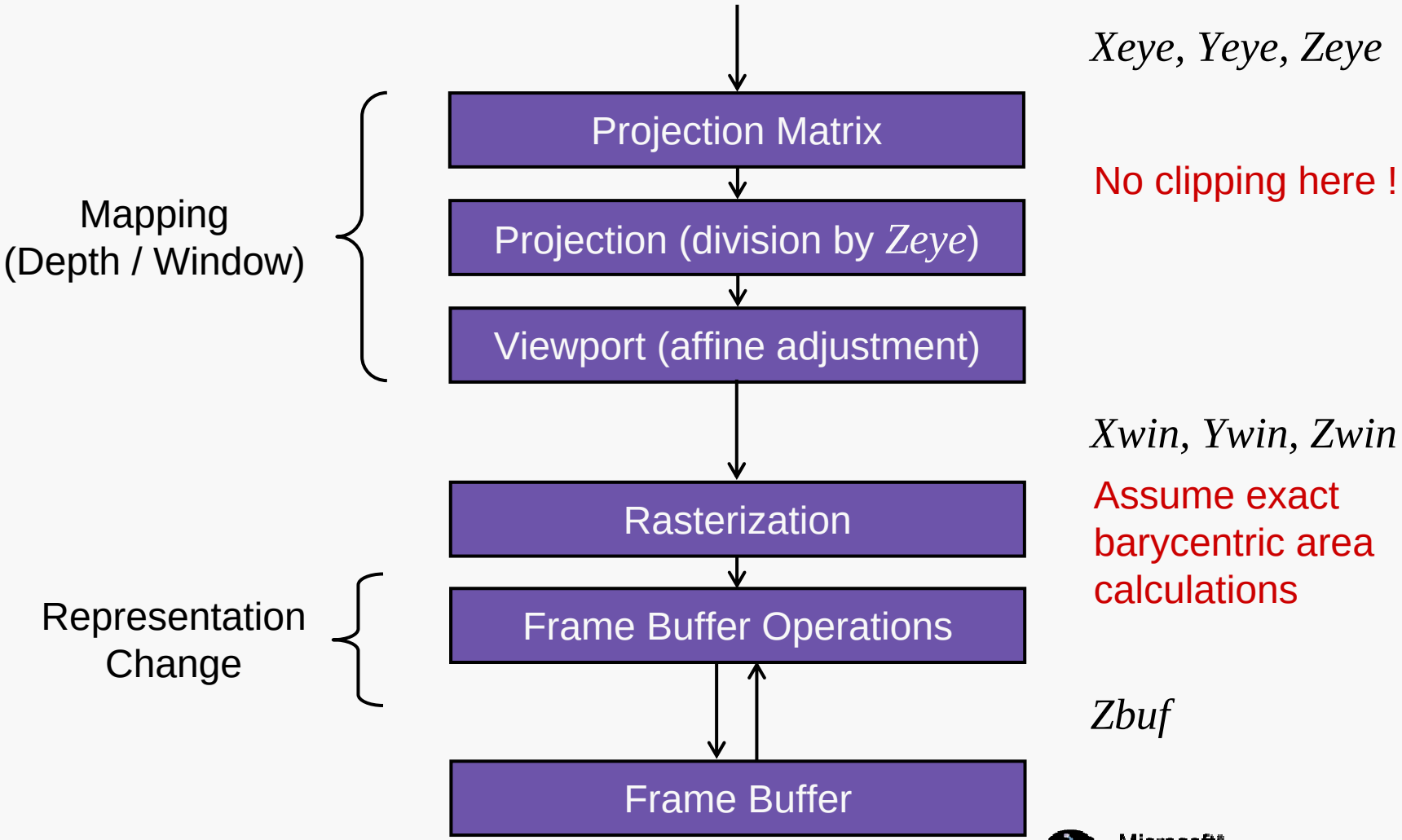

Trad Z-buffer (Wmap ideal, Zmap ideal, Ras ideal, Dcpl, Fov 90)
Zbuf (0.24)
Xwin, Ywin (10.8)
Combined (center)
Combined (corner)
Slopes are 2 and 1 respectively
FOV factor matters here
Smin is dominated by Xwin and Ywin representation here (explains comment in Lapidous01)
Bad
Good
Required separation (eye coordinates)
Zeye
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-4
10^-3
10^-2
10^-1
10^0


Trad Z-buffer (Wmap ideal, Zmap ideal, Ras ideal, Dcpl, Fov 90)
Zbuf (0.24)
Xwin, Ywin (10.8)
Combined (center)
Combined (corner)
Simulated (corner)
Slopes are 2 and 1 respectively
FOV factor matters here
Smin is dominated by Xwin and Ywin representation here (explains comment in Lapidous01)
Bad
Good
Simulation results in blue
Required separation (eye coordinates)
Zeye
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-4
10^-3
10^-2
10^-1
10^0

Compute *Smin* again

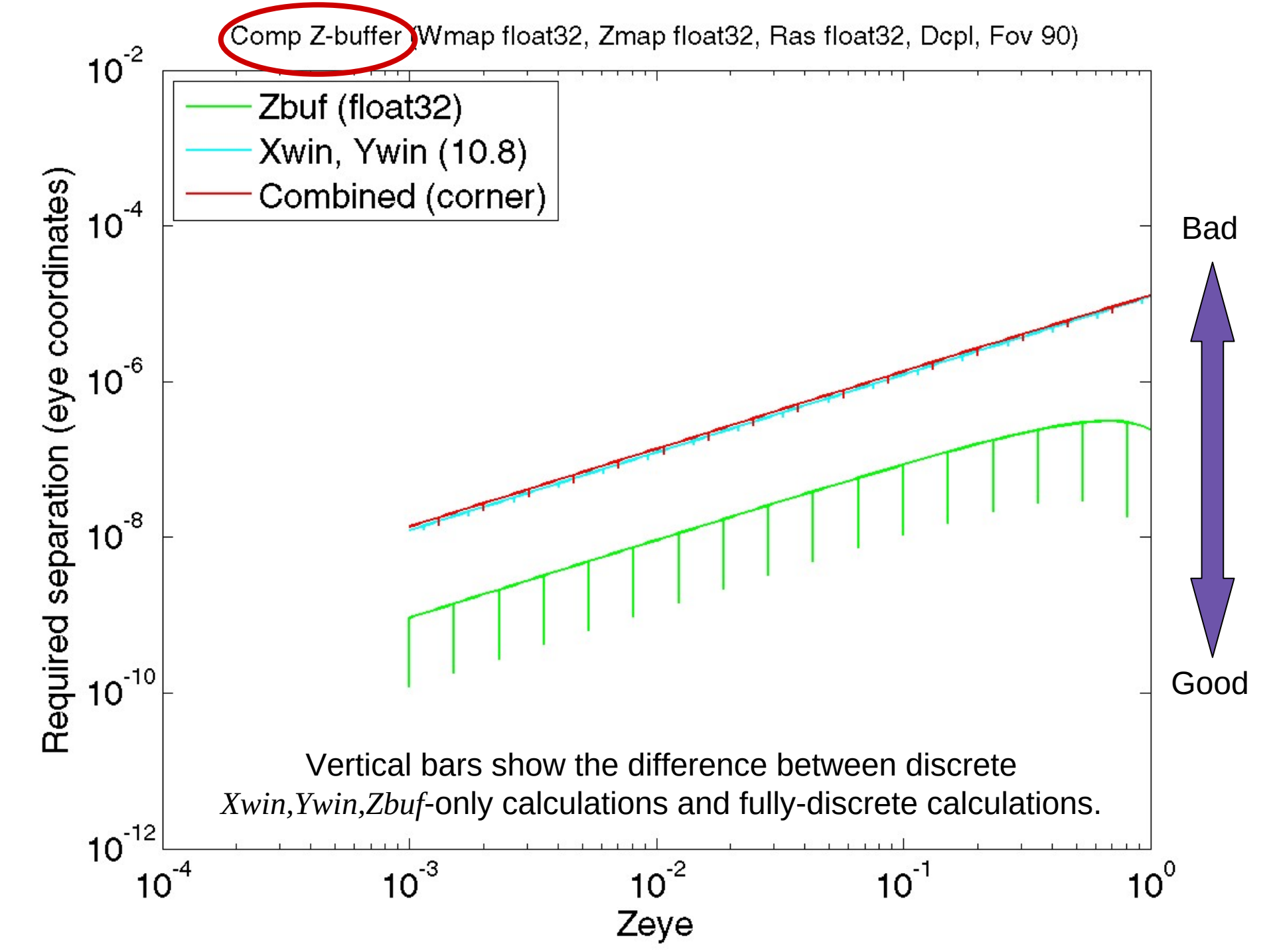
(all values treated as discrete)

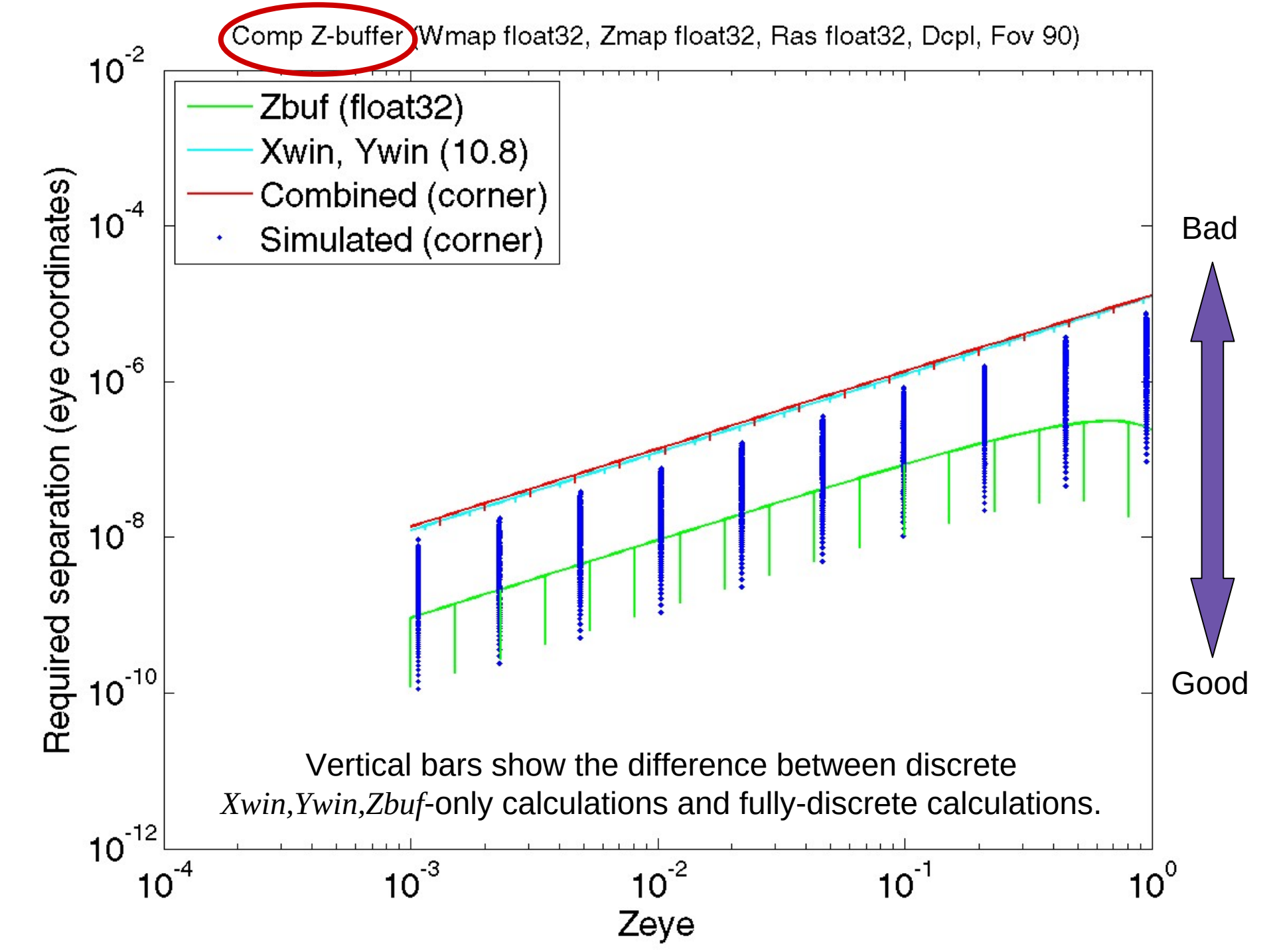
Cumulative Uncertainty Arithmetic

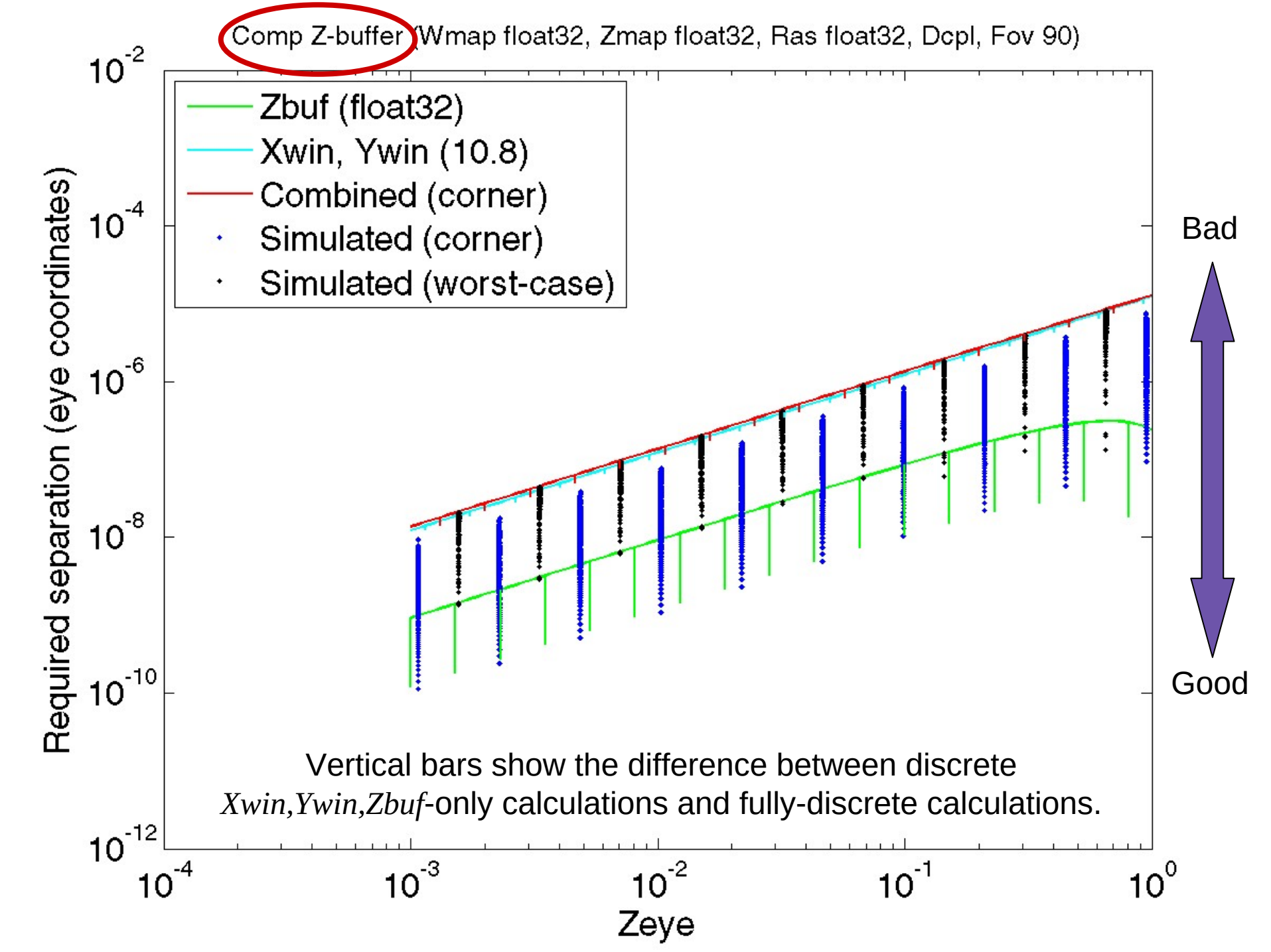


Interval Analysis

- We wrote our own interval analysis tool
- API allows specification of
  - Equation
  - Representations of operands and intermediates
  - Values of operands
- Evaluation is done using “ideal” arithmetic
  - Current ideal is IEEE 64-bit
- Handles polynomials correctly, e.g. $Zwin = \frac{C \cdot Zeye + D}{Zeye}$
- Reverse map to eye coordinates

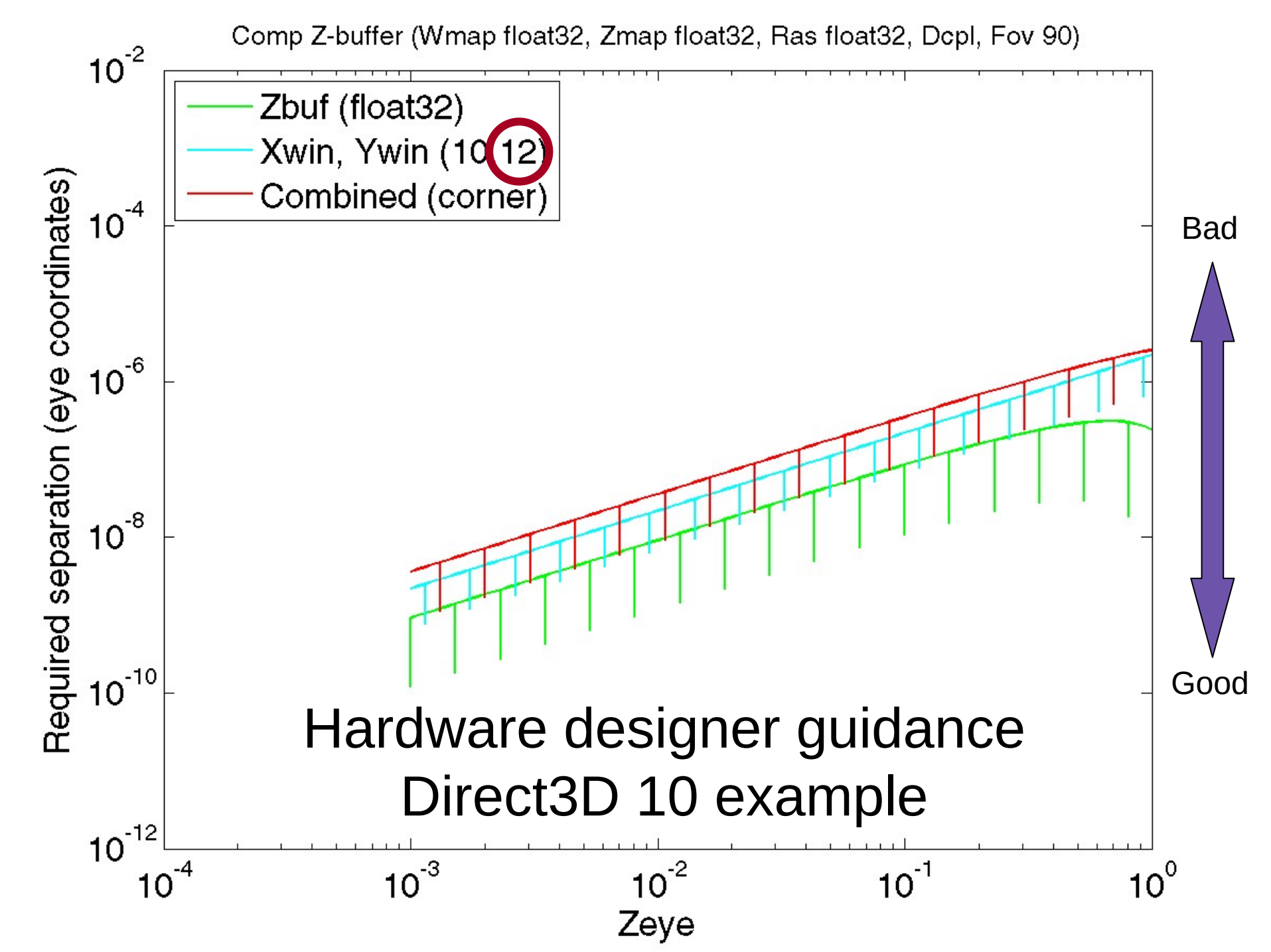

Comp Z-buffer (Wmap float32, Zmap float32, Ras float32, Dcpl, Fov 90)
Zbuf (float32)
Xwin, Ywin (10.8)
Combined (corner)
Required separation (eye coordinates)
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-4
10^-3
10^-2
10^-1
10^0
Zeye
Bad
Good
Vertical bars show the difference between discrete
Xwin,Ywin,Zbuf-only calculations and fully-discrete calculations.


Comp Z-buffer (Wmap float32, Zmap float32, Ras float32, Dcpl, Fov 90)
Zbuf (float32)
Xwin, Ywin (10.8)
Combined (corner)
Simulated (corner)
Required separation (eye coordinates)
Zeye
Bad
Good
Vertical bars show the difference between discrete
Xwin,Ywin,Zbuf-only calculations and fully-discrete calculations.


Comp Z-buffer (Wmap float32, Zmap float32, Ras float32, Dcpl, Fov 90)
Zbuf (float32)
Xwin, Ywin (10.8)
Combined (corner)
Simulated (corner)
Simulated (worst-case)
Required separation (eye coordinates)
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-4
10^-3
10^-2
10^-1
10^0
Zeye
Bad
Good
Vertical bars show the difference between discrete
Xwin,Ywin,Zbuf-only calculations and fully-discrete calculations.

Guidance

(for hardware designers
and application developers)


Comp Z-buffer (Wmap float32, Zmap float32, Ras float32, Dcpl, Fov 90)
Zbuf (float32)
Xwin, Ywin (10 12)
Combined (corner)
Required separation (eye coordinates)
Zeye
Bad
Good
Hardware designer guidance
Direct3D 10 example

Application Developer Guidance

- Can specify a tight *Smin* bound for a specific GPU !
- Linear increase of *Smin* with *Zeye*
  - Assume (near linear) complementary z-buffer
  - Matches linear *Xwin*, *Ywin* behavior
  - Is consistent with geometric LOD

$$S_{min} = K_{sep} Z_{eye}$$

Field of View

Viewport Size	30	60	90
512			
1024			
2048			

Conclusion

- Uncertainty analysis gives nice understanding of *Smin*
  - Window-coordinate representation matters
  - Wide FOV can increase *Smin* by 2x to 3x
- With this understanding, we can give guidance to
  - Hardware designers
  - Application developers (with community agreement)
- In the future, we could
  - Analyze contribution of clipping arithmetic
  - Better understand worst-case interval evaluation
  - Extend to new areas (e.g., shadow generation)

Thanks to

- Jonathan Su, my intern at MSR Asia
- David Blythe and Amar Patel of the DX10 team
- Marcel Gavriliu and Jim Kajiya, for help with interval analysis
- Turner Whitted and Harry Shum, for allowing me to do this research
- The Eurographics Hardware Workshop, now Graphics Hardware, for inviting me to speak in 1991