Half-Precision Floating-Point Ray Traversal
Matias Koskela, Timo Viitanen, Pekka Jääskeläinen & Jarmo Takala
Tampere University of Technology, Finland

The Proposed Ray Traversal Algorithm

Similar to (Keely, 2014), but can be done in existing half-precision hardware.

Ray traversal inner loop:
if node is an inner node
  Compute ray-AABB intersection tests in the child's hierarchical coordinates
  in half-precision floating-point numbers
  For every intersected child
    Move ray origin to edges of intersected child bounding boxes
    Convert ray origin to hierarchical child coordinates
  end for
else (node is a leaf node)
  Do triangle tests
end if
end loop

Conversion to Hierarchical Coordinates

The proposed method scales all hierarchical coordinates according to largest bounding box side dimension
- This way rays are not bending when they go deeper into hierarchy
- But precision is lost
Only the interval of -16384 ... 16384 is used to keep ray bounding box tests from rounding up to infinity (fourth of the maximum half-precision non-infinite interval -65504 ... 65504)

Hierarchical Bounding Volume Hierarchies

- Used in previous work (Mahovsky and Wyvill 2006, Keely 2014, Koskela & al 2015)
- Allows tighter bounding boxes with less accurate datatypes

Ray Origin Movement

Origin is moved to the edge of every intersected child node’s AABB
- This way origin stays within the half-precision non-infinite range even though it is converted to the child node’s coordinates
- Requires extra computations & extra memory space for the intermediate values

Results

Average differences to regular single-precision traversal show that proposed method gets better when wider vectors are used
- With narrow vectors the benefits are hidden by reduced precision bounding volumes, extra computations and extra stacks for intermediate values
- If a ray tracer is fastest with wide vector instructions and the targeted hardware has native half-precision computation support the proposed ray traversal should be testing.

References: