

Supplemental Document

Full Experiment Results for Paper: “Ray Specialized Contraction on Bounding Volume Hierarchies”

Yan Gu Yong He Guy E. Blelloch

Carnegie Mellon University

This supplemental document provides full experiment results for the paper submission entitled “Ray Specialized Contraction on Bounding Volume Hierarchies”. Due to the page limit, only part of the experiment can be included in the paper.

1. Experiment Setup

The evaluation in our paper focuses on both the number of ray-box tests, and the actual execution time on a many-core CPU. We claim that the number of ray-box tests is a good indicator for BVH quality, because this number does not depend on details of the hardware or implementation, and can be easily reproduced. Since we cannot implement all of the state-of-the-art ray tracers on different platforms, we report our actual running time on a 40-core machine with four 10-core Intel E7-8870 Xeon processors (1066 MHz bus). Parallel implementations were compiled with CilkPlus, which is included in G++. Our ray tracing code is similar to some of the recent works [BWB08, Tsa09]. The tracing speed is provided in Table 2 and 3. More implementation details and analysis can be found in Section 3. Notice that tree structure needs to be rearranged for tracing n -ary BVH to utilize AVX. Since our algorithm requires to slightly change the pipeline of ray tracing, we cannot directly use some existing systems like Embree [WWB*14] or OptiX [PBD*10] and report experiment results. Nevertheless, there are two reasons that we still believe our algorithm is meaningful: first, our algorithm is simple, so that it will not be hard for the software engineers in their groups to integrate into these systems; second, we explain the reason that tracing our new BVHs are even more efficient in Section 3 due to the special properties of our contracted BVHs, even if the traversal code is less “highly” hand-tuned.

We use 15 test scenes in our experiments, which contain significant scene-to-scene variations. Our method tends to reduce ray-box tests due to BVH imbalances in complex geometry models, so we mainly focus on 10 real-world scenes, which include: 3 widely used architectural models CONFERENCE, CRYTEK-SPONZA and SAN-MIGUEL; a complex building SODA-HALL to be rendered separately inside and outside; 2 city models ARABIC and BABYLONIAN from the Mitsuba distribution [Jak10] showing large spatial extends; and 3 game scenes TRAIN-STATION, EPISODE2 and WAREHOUSE from HalfLife2, with complex geometry. Experimental results for the other 5 scenes, mainly objects or scanned models including BUDDHA, HAIRBALL, FAIRY, DABROVIC-SPONZA and POWERPLANT, are given in the supplemental material in detail, with a brief abstract shown in Table 9.

We show the benefits of our method by studying the performance improvement based on starting with the BVH constructed by three different algorithms: a top-down full-sweep SAH build (short for SAH) [GS87], a bottom-up build using approximate agglomerative clustering with HQ parameters (short for AAC) [GHFB13], and a top-down build using spatial splits with default parameters (short for SBVH) [SFD09]. These algorithms generate high-quality BVHs using different approaches, so the evaluation results are representative. Renderings use 32 probe rays (diffuse bounce rays) per pixel and one to several area light sources depending on scene complexity. We only use 32 probe rays per pixel because more probe rays lead to more sample rays in overall, which creates a more accurate estimation for Pass-test rate in a node and higher BVH quality after contraction. Here we show that only such limited number (32) of probe rays is sufficient to provide a considerable speedup. More than 5 camera positions for outdoor scenes and 3-5 for indoor scenes are used, and the results are averaged. We further show in Table 6 that the benefit of the new generated BVH is actually insensitive to different camera positions. We pre-render 1 pixel per 16×16 block in screen space, and use these sample rays to generate statistics on the BVH. Our experiment shows that the threshold t in *StopCriterion* in RDTC is insensitive, and in the experiments we use the maximum number of rays for a single sample pixel.

In our experiment, we extensively use the “relative ratio” or “relative performance” to show the acceleration of our approach, and here it is defined as the total amount of work (number of ray-box tests or wall clock time) which is done on the contracted BVH divided by that on the original binary BVH.

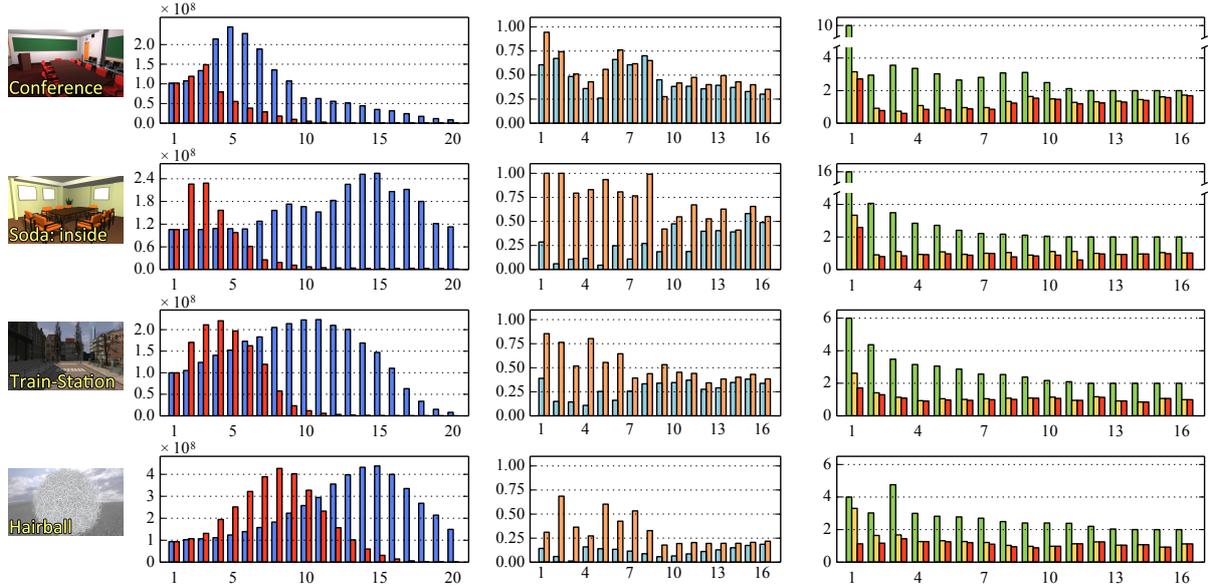
2. Scene-by-Scene Acceleration

| Scene | Initial BVH type | # ray-box tests non-opt BVH diff / shad | Rel. # ray-box tests | | Relative runtime | | BVH imbalance | | Contracted nodes | | Rel. node depth | Ave. num. of branch / pass |
|--|------------------|---|----------------------|------------------|--------------------|--------------------|---------------|------|------------------|------|-----------------|----------------------------|
| | | | SATC diff / shad | RDTC diff / shad | Single diff / shad | Packet diff / shad | SATC | RDTC | num. | pct. | | |
|  Conference | SAH | 42.9 / 31.3 | 0.75 / 0.72 | 0.71 / 0.67 | 0.74 / 0.69 | 0.76 / 0.70 | 0.51 | 0.58 | 1.9K | 1.1% | 0.31 | 5.2 / 1.3 |
| | AAC | 35.0 / 24.9 | 0.76 / 0.72 | 0.73 / 0.71 | 0.76 / 0.75 | 0.79 / 0.77 | 0.24 | 0.39 | 1.8K | 1.5% | 0.36 | 4.6 / 1.2 |
| | SBVH | 37.4 / 28.9 | 0.79 / 0.73 | 0.75 / 0.66 | 0.78 / 0.69 | 0.80 / 0.69 | 0.49 | 0.61 | 2.3K | 1.1% | 0.30 | 5.3 / 1.5 |
|  Crytek-Sponza | SAH | 105.4 / 86.1 | 0.80 / 0.56 | 0.72 / 0.50 | 0.75 / 0.56 | 0.77 / 0.54 | 0.37 | 0.49 | 5.6K | 3.6% | 0.30 | 6.1 / 1.5 |
| | AAC | 87.5 / 54.8 | 0.82 / 0.66 | 0.74 / 0.55 | 0.71 / 0.61 | 0.70 / 0.59 | 0.21 | 0.44 | 5.2K | 4.8% | 0.25 | 5.2 / 1.3 |
| | SBVH | 71.6 / 51.7 | 0.88 / 0.85 | 0.81 / 0.57 | 0.83 / 0.65 | 0.85 / 0.61 | 0.30 | 0.45 | 5.9K | 2.8% | 0.34 | 4.6 / 1.2 |
|  Soda: inside | SAH | 68.5 / 40.3 | 0.90 / 0.70 | 0.69 / 0.46 | 0.70 / 0.54 | 0.73 / 0.53 | 0.32 | 0.65 | 1.2K | 0.1% | 0.15 | 6.6 / 1.2 |
| | AAC | 139.9 / 65.3 | 0.76 / 0.77 | 0.46 / 0.48 | 0.43 / 0.55 | 0.42 / 0.57 | 0.20 | 0.45 | 4.7K | 0.6% | 0.15 | 7.0 / 1.2 |
| | SBVH | 67.0 / 50.1 | 0.95 / 0.78 | 0.70 / 0.58 | 0.66 / 0.65 | 0.73 / 0.65 | 0.26 | 0.61 | 1.4K | 0.1% | 0.23 | 5.3 / 1.1 |
|  Soda: outside | SAH | 44.9 / 30.7 | 0.87 / 0.76 | 0.76 / 0.68 | 0.83 / 0.69 | 0.76 / 0.70 | 0.37 | 0.56 | 3.2K | 0.2% | 0.30 | 4.9 / 1.1 |
| | AAC | 54.0 / 36.7 | 0.83 / 0.82 | 0.75 / 0.73 | 0.73 / 0.68 | 0.73 / 0.74 | 0.20 | 0.42 | 3.7K | 0.4% | 0.33 | 4.2 / 1.2 |
| | SBVH | 34.8 / 27.6 | 0.97 / 0.86 | 0.80 / 0.68 | 0.90 / 0.68 | 0.83 / 0.71 | 0.21 | 0.50 | 3.6K | 0.2% | 0.43 | 3.8 / 1.0 |
|  Arabic | SAH | 88.5 / 53.7 | 0.97 / 0.98 | 0.76 / 0.69 | 0.83 / 0.77 | 0.80 / 0.72 | 0.36 | 0.49 | 4.7K | 1.7% | 0.33 | 4.5 / 1.3 |
| | AAC | 75.0 / 44.6 | 0.85 / 0.85 | 0.76 / 0.73 | 0.82 / 0.81 | 0.84 / 0.81 | 0.20 | 0.45 | 3.8K | 2.1% | 0.35 | 4.4 / 1.3 |
| | SBVH | 50.9 / 38.3 | 0.92 / 0.87 | 0.79 / 0.75 | 0.84 / 0.82 | 0.86 / 0.84 | 0.24 | 0.48 | 5.8K | 1.5% | 0.37 | 4.0 / 1.1 |
|  Babylonian | SAH | 61.0 / 39.2 | 0.88 / 0.84 | 0.77 / 0.68 | 0.79 / 0.73 | 0.85 / 0.76 | 0.34 | 0.51 | 2.2K | 0.7% | 0.34 | 4.2 / 1.1 |
| | AAC | 65.4 / 43.9 | 0.83 / 0.72 | 0.77 / 0.76 | 0.73 / 0.77 | 0.80 / 0.74 | 0.20 | 0.46 | 2.3K | 1.0% | 0.24 | 5.8 / 1.4 |
| | SBVH | 45.1 / 30.0 | 0.92 / 0.93 | 0.80 / 0.71 | 0.78 / 0.78 | 0.78 / 0.80 | 0.26 | 0.57 | 2.5K | 0.5% | 0.34 | 4.1 / 1.0 |
|  Train-Station | SAH | 64.1 / 39.2 | 0.89 / 0.86 | 0.79 / 0.71 | 0.75 / 0.77 | 0.80 / 0.70 | 0.29 | 0.49 | 2.0K | 1.7% | 0.34 | 4.0 / 1.2 |
| | AAC | 65.1 / 37.3 | 0.93 / 0.96 | 0.78 / 0.68 | 0.77 / 0.75 | 0.82 / 0.81 | 0.17 | 0.41 | 1.8K | 2.3% | 0.31 | 4.1 / 1.2 |
| | SBVH | 57.1 / 37.2 | 0.91 / 0.76 | 0.80 / 0.69 | 0.84 / 0.72 | 0.79 / 0.72 | 0.25 | 0.48 | 2.4K | 1.7% | 0.39 | 3.9 / 1.1 |
|  Episode2 | SAH | 72.9 / 47.8 | 0.91 / 0.91 | 0.74 / 0.64 | 0.67 / 0.59 | 0.72 / 0.62 | 0.27 | 0.56 | 2.9K | 0.4% | 0.21 | 5.3 / 1.3 |
| | AAC | 74.3 / 48.2 | 0.93 / 0.93 | 0.74 / 0.72 | 0.75 / 0.70 | 0.76 / 0.68 | 0.17 | 0.56 | 2.6K | 0.6% | 0.17 | 5.5 / 1.3 |
| | SBVH | 68.4 / 42.6 | 0.95 / 0.95 | 0.74 / 0.66 | 0.78 / 0.71 | 0.82 / 0.68 | 0.25 | 0.62 | 3.1K | 0.4% | 0.20 | 5.1 / 1.2 |
|  Warehouse | SAH | 73.5 / 57.5 | 0.93 / 1.08 | 0.68 / 0.65 | 0.67 / 0.58 | 0.65 / 0.64 | 0.30 | 0.59 | 3.0K | 2.2% | 0.29 | 4.9 / 1.3 |
| | AAC | 72.1 / 55.8 | 0.84 / 0.84 | 0.67 / 0.63 | 0.68 / 0.66 | 0.68 / 0.68 | 0.20 | 0.57 | 2.4K | 3.1% | 0.23 | 5.0 / 1.3 |
| | SBVH | 58.5 / 54.7 | 0.94 / 0.92 | 0.75 / 0.68 | 0.82 / 0.70 | 0.80 / 0.69 | 0.28 | 0.58 | 3.6K | 2.1% | 0.35 | 4.0 / 1.2 |
|  San-Miguel | SAH | 142.7 / 67.0 | 0.89 / 0.72 | 0.80 / 0.75 | 0.87 / 0.74 | 0.90 / 0.70 | 0.28 | 0.37 | 14.4K | 0.3% | 0.46 | 3.9 / 1.2 |
| | AAC | 143.0 / 60.8 | 0.85 / 0.88 | 0.79 / 0.81 | 0.82 / 0.87 | 0.88 / 0.78 | 0.18 | 0.32 | 14.9K | 0.4% | 0.43 | 3.9 / 1.2 |
| | SBVH | 106.2 / 55.9 | 0.94 / 0.75 | 0.83 / 0.67 | 0.91 / 0.71 | 0.90 / 0.69 | 0.25 | 0.37 | 13.1K | 0.2% | 0.51 | 3.7 / 1.2 |
| Average 10 | SAH | | 0.88 / 0.81 | 0.75 / 0.64 | 0.76 / 0.66 | 0.77 / 0.66 | | | | | 0.30 | 4.9 / 1.2 |
| | AAC | | 0.84 / 0.82 | 0.72 / 0.68 | 0.71 / 0.71 | 0.74 / 0.72 | | | | | 0.28 | 5.0 / 1.3 |
| | SBVH | | 0.92 / 0.85 | 0.78 / 0.66 | 0.80 / 0.70 | 0.81 / 0.70 | | | | | 0.35 | 4.4 / 1.2 |

Table 1: Detail experimental results for different scenes with various initial BVHs. Results for numbers of ray-box tests for non-optimized BVHs, relative ratios of ray-box test for both SATC and RDTC comparing to non-optimized BVHs, relative ratios on runtime for actual wall-clock time for ray-primitive testing for RDTC on both single and packet ray tracing (actual running time will be provided in Table 2 and Table 3) BVH imbalance descriptors for both SATC and RDTC, reconstructed BVH nodes for RDTC, relative node depth to reach triangles for ray-box testing between RDTC and initial BVH, and average numbers of branches and Pass-tests for new contracted node are provided. The data in the last column are averaged on the weight of the VisitCount in each node. “diff / shad” means diffuse rays (and other rays that query for the first intersection) / shadow rays.

To start with, we first analyze the improvement of performance by BVH contraction on different scenes. Table 10 compares the relative performance based on different parameters, with both number of ray-box tests (for SATC and RDTC) and wall clock time (for RDTC). The table also provides the tree imbalance, number of contracted nodes for RDTC, relative node depth, and average number of branches for new generated node. All these data are generated by single ray tracing, but the running time for packet ray tracing is also provided, which shows a similar speedup.

The SATC heuristic which tries to avoid unnecessary ray-box tests caused by structural imbalance, can reduce the tests by



(a) node traversed, by original BVH (b) structural (cyan) and overall (o- (c) Average number of branches (green), intersected boxes (yellow) and actual traversed nodes (red)

Figure 1: Detail results by levels in the original and contracted BVH. Column (a) shows the nodes that actually traversed (i.e. by Pass-tests) in each level. Column (b) provides structural and overall imbalance descriptors defined in Section 2. Column (c) gives the average number of branches, intersected boxes and actual traversed nodes on contracted BVH, and each node is weighted by the number in its counter. Initial BVHs are top-down SAH BVHs.

up to 25% on diffuse rays and 45% on shadow rays (column SATC in “Rel. # ray-box tests”). However, this number varies significantly across the scenes and BVH construction methods, and can even be negative. The RDTC heuristic however, captures inefficiencies due to both structural and ray-distribution imbalance, gets a consistent improvement of 20-30% (average 25%) for diffuse rays, and 25-55% (average 35%) for shadow rays (column RDTC in “Rel. # ray-box tests”). Similar improvements in runtime are also observed. Moreover, these improvements are less related to BVH construction approaches, but are more scene depended. Such reductions in the number of ray-box tests for ray tracing are significant since the ray-primitive testing has logarithmic time complexity.

To further demonstrate the inefficiency in BVH traversal, for a set of nodes S , we define the following “imbalance descriptor” (Imb) to measure the difference in probability for traversing the subtrees:

$$Imb(S) = \frac{\sum_{s \in S} (visitCount(s) |\alpha_{s,left} - \alpha_{s,right}|)}{\sum_{s \in S} visitCount(s)} \quad (1)$$

where α is measured by different parameters of the BVH contraction. The argument S for Imb in this paper can be the set of all the nodes in a BVH (in Table 10), or the nodes in a specific level (in Figure 1), and the value range is between 0 and 1. As we discussed previously, an ideal data structure should have a small value of Imb . We claim that this function predicts the improvement by our method very well, and the linear regression between them are shown in Section 6.

The number of contracted nodes for RDTC is provided, and usually only a few thousand tree nodes (1.2K to 5.9K, except for San-Miguel which contains 8M triangles) are reconstructed in our methods. The consumed time for BVH contraction is very short (usually less than 1ms), and about the time to trace a few hundred rays. Therefore, it is affordable to run the BVH contraction algorithm on **every** frame. We tested the hybrid parameters for SATC and RDTC to a full BVH contraction (i.e. to use α_N in CBTC), and the difference between the hybrid parameters and RDTC in relative ratio in ray-box tests is less than 1% in all scenes with any initial BVH. Hence, we believe that only this small fraction of the tree (the contracted part, 0.1% to 3% of overall tree nodes) covers most of the structure and ray distribution imbalance. Meanwhile, Table 10 also shows that an average of 8-15% improvement on diffuse rays and 13-18% on shadow rays is caused by structural imbalance and caught by SATC, and an extra 12-14% and 13-20% improvement is caused by extra ray-distribution imbalance and caught by RDTC.

We investigate the benefits of our method by further looking at four representative scenes in Figure 1, which are architectural

model CONFERENCE with mainly structural imbalance, building SODA-HALL with imbalance in ray distribution, game scene TRAIN-STATION with imbalance on both, and finely tessellated objects HAIRBALL that has a balanced initial BVH. All their traversal details are computed with an initial BVH generated by top-down SAH.

In column (a), bars indicate the number of Pass-tests in each level. Pass-tests should be avoided in ray-primitive testing because they cannot provide useful pruning during traversal and creates extra data accesses. As we anticipated, this number reduce by a factor of 60% to 80% on the first 3 scenes, and about 40% for balanced initial BVH.

Figure 1, Column (b) shows structural and ray distribution imbalances by levels, which are computed by the imbalance descriptor with α separately from SATC and RDTC. These figures show that most significant imbalance happens at the top (around 10) levels in the BVH, which is where our algorithm focuses on. This is further shown by column (c), which indicates that few contractions happen beyond the top levels since the average branches drop down to 2 quickly. Moreover, even if we have a multi-branch (up to 16 branches) BVH at the first several levels, the actual number of Pass-tests is relatively low (≤ 3.3 at root node, ≤ 1.6 in 2 to 10 levels, average 1.2 to 1.3 for contracted node as shown in Table 10), which means our BVH contraction will not require sorting many boxes to order them from front to back.

| Model | Diffuse rays | | | Shadow rays | | |
|---------------|--------------|------|------|-------------|------|------|
| | SAH | AAC | SBVH | SAH | AAC | SBVH |
| Conference | 40.2 | 56.3 | 47.3 | 57.2 | 80.4 | 66.3 |
| Crytek-Sponza | 20.7 | 25.5 | 31.4 | 28.7 | 34.8 | 38.9 |
| Soda: inside | 40.6 | 19.2 | 43.1 | 60.2 | 44.7 | 62.7 |
| Soda: outside | 57.1 | 48.8 | 73.5 | 76.2 | 66.3 | 90.0 |
| Arabic | 11.6 | 13.8 | 19.3 | 15.0 | 18.2 | 22.0 |
| Babylonian | 27.1 | 25.5 | 34.7 | 38.3 | 31.3 | 46.0 |
| Train-Station | 32.3 | 31.6 | 41.2 | 39.9 | 40.8 | 52.6 |
| Episode2 | 34.1 | 32.6 | 36.1 | 45.2 | 45.1 | 46.9 |
| Warehouse | 14.0 | 14.6 | 17.3 | 19.8 | 21.1 | 22.7 |
| San-Miguel | 8.8 | 8.9 | 11.4 | 10.3 | 10.9 | 12.9 |

Table 2: Performance of our ray tracer (million rays per second) on single ray tracing on non-contracted BVHs.

| Model | Diffuse rays | | | Shadow rays | | |
|---------------|--------------|------|------|-------------|-------|-------|
| | SAH | AAC | SBVH | SAH | AAC | SBVH |
| Conference | 50.0 | 73.9 | 53.5 | 68.6 | 100.4 | 83.1 |
| Crytek-Sponza | 24.8 | 32.6 | 40.8 | 34.8 | 40.5 | 49.5 |
| Soda: inside | 62.8 | 32.0 | 64.2 | 102.8 | 78.2 | 109.6 |
| Soda: outside | 59.1 | 58.2 | 78.6 | 78.6 | 71.1 | 108.3 |
| Arabic | 14.1 | 17.5 | 23.7 | 18.0 | 22.4 | 28.3 |
| Babylonian | 34.3 | 30.0 | 38.4 | 46.4 | 40.0 | 59.4 |
| Train-Station | 38.4 | 39.8 | 46.6 | 49.3 | 48.2 | 59.7 |
| Episode2 | 31.4 | 32.0 | 36.9 | 45.1 | 43.2 | 51.3 |
| Warehouse | 15.4 | 18.7 | 20.6 | 26.0 | 24.6 | 24.6 |
| San-Miguel | 9.2 | 9.4 | 11.9 | 11.6 | 12.4 | 13.0 |

Table 3: Performance of our ray tracer (million rays per second) on packet ray tracing on non-contracted BVHs.

3. Implementation Details on Traverse

In this section we illustrate the implementation details on BVH traverse on contracted BVHs.

The code to traversing binary BVH is usually highly optimized, including hand-tune operations, dedicate register allocation, etc. For contracted BVH, only the top levels are reconstructed, and the contracted flag of these nodes are marked as `True`. For non-contracted node, the highly-optimized code for binary BVH traversing is still able to use, because the whole subtree is not changed. For contracted node, an extra loop variable and a more complex sorting process are needed. However, we claim that the extra steps will not affect the running speed.

For non-contracted nodes, multiple box-tests occur together (average 4.4-5.0 shown in Table 10). Nevertheless, since the contraction reduces Pass-tests, only average 1.2 to 1.3 Pass-tests occur on each node. Since only a small fraction (about 25%) of

Pass-tests are on each node, sorting the children from-to-end is cheap, since we are usually sorting no more than 2 elements except for the root node (Column (c) in Figure 1). Moreover, the nodes that need sorting are much less and reduced by about 70% on average (Column “relative node depth” in Table 10). Overall, the time spend in sorting process is actually faster than that without BVH contraction. Furthermore, The path to reach triangles are much shortened (average length of 2.6 to reach triangles), comparing to a path with usually 6-15 levels to reach the corresponding BVH nodes. The reduction on average depth can largely speedup the traversing process on both stack and stackless implementation, and overcome the extra cost to use the loop variable.

The experiment results in Table 10 and Figure 1 shows that traversing the new contracted multi-branch BVHs will not cause inefficiency comparing to binary BVHs, since the relative ratio of actual running times is similar to the ratio of decreased ray-box tests.

4. Evaluation on Details

From the previous section we show that the improvement on overall performance on contracted BVHs. In this section, we use control variable method to show the impacts on different rendering settings, including sample size, lighting environment, camera position, and the order of reflection rays. We choose the model TRAIN-STATION as the representative example to provide experimental results, because this model has a moderate geometry complexity and spatial extent (mainly consisting of a building and a square).

4.1. Sample Size

We first analyze the impact on different sample sizes and overall speedups. Intuitively, more samples provide better estimation on the probabilities of Pass-tests, but decrease the number of rays that traced by the new contracted BVH. Hence, there exists a balance and optimal point for the sample size. Table 4 provides relative ray-box testing ratios for different sample sizes. We can find that only a negligible change is in relative ratios to traverse non-sample rays when sample size is between 1 and 32. Therefore, we want to have less samples so that more pixels that can be traced with contracted BVHs (no need for the sample pixels to be computed again since the associated rays are already traced), and a sample size between 8 to 32 usually provides the best overall improvement. In practice, one thousand sample pixels are sufficient for our approach.

| Sample size | | 1 | 4 | 8 | 16 | 32 | 48 | 64 |
|-------------|---------|------|------|------|------|------|------|------|
| diffuse | opt. | .790 | .791 | .791 | .793 | .794 | .804 | .812 |
| | overall | 1.00 | .804 | .795 | .794 | .795 | .804 | .812 |
| shadow | opt. | .707 | .710 | .710 | .711 | .712 | .720 | .729 |
| | overall | 1.00 | .728 | .714 | .712 | .712 | .720 | .729 |

Table 4: Relative ratios of the number of ray-box tests with RDTC contraction for different sample sizes, on scene TRAIN-STATION with initial SAH BVHs. Sample size of P indicates that we pre-render 1 pixel per $P \times P$ block in screen space. The optimized (opt.) rows represent the relative ratios of the number of ray-box tests across all rays (including the sample) on contracted BVHs using these sample, comparing with non-contracted BVHs. The overall rows represent the relative ratios on overall cost, including sample pixels using non-contracted BVHs and other pixels using contracted BVHs. (The overall relative ratio a can approximately be estimated using optimized relative ratio o by a weighted average of $a \approx (1 - P^{-2})o + P^{-2}$, but here we provided actual data from our ray tracer.)

4.2. Lighting Setting, Camera Position, and Higher-Order Diffuse-Bouncing Rays

Lighting setting. Now we try to illustrate the relationship between light environments and performance of contracted BVHs. We set the light environments in TRAIN-STATION to be one or multiple area light sources that create different ratios of occluded shadow rays. This may change the performance of contracted BVHs because once we find an occlusion for a shadow ray, the traverse function will have an early exit (line 8 in Algorithm 2), which will increase the number of Prune-tests since the rest parts of the BVH does not need to be traversed. Therefore, higher ratios of occluded shadow rays can accelerate the traversing speed for shadow rays since intersections are found faster after contraction. However, the performance for diffuse rays should not be affected by light environments.

Table 5 shows the performance of our approach with different light environments. We put several area light sources in the scene and the rendered images and relative ratios of ray-box testing numbers are provided. As we expected, the performance to traverse shadow rays on contracted BVHs get better improvement when a higher ratio of occluded rays appear. Moreover, better coherence of shadow rays (in first 3 images) generally improves the performance of our approach.

camera positions. We then show that different camera positions do not impact the performance of our approach much, and the

| Image |  |  |  |  |
|-----------|---|---|---|--|
| Occ ratio | 32% | 46% | 92% | 56% |
| SAH | 0.79 / 0.72 | 0.79 / 0.71 | 0.79 / 0.60 | 0.79 / 0.77 |
| AAC | 0.78 / 0.70 | 0.78 / 0.68 | 0.79 / 0.58 | 0.79 / 0.68 |
| SBVH | 0.80 / 0.71 | 0.80 / 0.69 | 0.80 / 0.64 | 0.81 / 0.74 |

Table 5: Relative ratios of ray-box tests (diffuse ray / shadow ray) by different light sources with RDTC, on scene TRAIN-STATION with initial SAH BVHs. 1 area light source is placed in different directions for the first 3 images, and all 3 are placed in the last image. The row “occ ratio” provides the percentage of occluded rays (i.e., at least one intersection along the ray) among all shadow rays.

associated data are shown in Table 6. High quality ray tracing requires numerous diffuse rays to generate global illumination effects. Hence such a large number of incoherent rays distribute fairly randomly no matter where the camera positions are (except for the last case that is in a certain isolated room). A difference of only 5% is seen between the extreme cases (most visible primitives in the first image versus least in the third image) viewed within the same connected volume in the scene.

| Image |  |  |  |  |
|-------|---|---|---|--|
| SAH | 0.79 / 0.77 | 0.78 / 0.75 | 0.76 / 0.71 | 0.77 / 0.69 |
| AAC | 0.78 / 0.77 | 0.77 / 0.76 | 0.73 / 0.73 | 0.69 / 0.65 |
| SBVH | 0.80 / 0.77 | 0.79 / 0.76 | 0.76 / 0.76 | 0.75 / 0.71 |

Table 6: Relative ratios of ray-box tests (diffuse bouncing ray / ambient occlusion ray) by different camera positions with RDTC, on scene TRAIN-STATION with initial SAH BVHs. Various camera positions are used to create different images: different sides of the square in the first two images, a close look to a corner in the third image, and inside the building in the last image.

higher-order diffuse-bouncing rays. In previous experiments, we only traced depth-1 diffuse-bouncing rays. Now we show that performance improvements are also achieved with higher-order diffuse-bouncing rays. Numbers of ray-box tests with binary BVHs and relative ratios with contracted BVHs for 3 different scenes with 1st, 2nd and 3rd diffuse-bouncing rays are shown in Table 7. The results indicate that the acceleration by BVH contraction is insensitive to higher-order bouncing rays (1% to 3% difference in relative ratios).

| Scene | 1st bouncing | 2nd bouncing | 3rd bouncing |
|---------------|--------------|--------------|--------------|
| Conference | 42.87 / 0.71 | 45.78 / 0.73 | 47.74 / 0.74 |
| Train-Station | 64.12 / 0.79 | 66.07 / 0.80 | 67.39 / 0.80 |
| Soda: inside | 68.50 / 0.69 | 64.35 / 0.70 | 64.10 / 0.71 |

Table 7: Numbers of ray-box tests for non-optimized BVH and relative ratios by RDTC, for different order diffuse-bouncing rays on scene TRAIN-STATION. Original BVHs and contractions are based on BVHs generated by the top-down full-sweep SAH.

4.3. N -ary BVHs

There are two major ways to utilize the SIMD units on either CPU or GPU. The previous experiments focus on testing one bounding box versus multiple rays, i.e. packet ray tracing. Here we provide some experimental results to show the improvement of our approach when applied to n -ary BVHs, so that multiple (usually 4) bounding boxes can simultaneously test together. We run experiments on three different n -ary BVH construction methods: directly collapsing [DHK08] (direct collapse column), only by surface area [WBB08] (CBTC-SA column), and by both ray distribution and surface area (BCTC-RD column, our method).

Experiment results on 10 outdoor scenes combining with 3 initial BVH construction algorithms are shown in Table 11. We can find that CBTC-RD provides a 15% / 21% (diffuse / shadow ray) improvement for Quad-BVH and 25% / 30% for Oct-BVH compared with direct collapsing, and 10% / 12% for Quad-BVH and 14% / 15% for Oct-BVH compared with CBTC-SA. Notice that this improvement is irrelevant to implementation of the ray tracer since we can use the same versions of code to run ray-primitive tests on our new approach CBTC-RD as they previously designed and optimized to trace direct collapse BVHs and CBTC-SA BVHs.

The improvement for Quad-BVH is less due to the limited possibilities for modifications. To generate statistics for all BVH nodes as the input for BVH contraction, the initial BVH needs to be a binary BVH. Nevertheless, as we discussed in Section 4.1, it is sufficient to only sample about 0.1% pixels, so the cost in this step is negligible.

Moreover, it is interesting to point out that, state-of-the-art CPU ray tracing kernels (like Embree for Intel CPUs) usually use Quad-BVHs, because tree quality decreases significantly on wider-branch BVHs (average 30% extra ray-box tests on diffuse

| Scene | Initial BVH type | # box testing direct collapse diff. / shad. | Quad-BVH relative ratios | | # box testing direct collapse diff. / shad. | Oct-BVH relative ratios | |
|---|------------------|---|--------------------------|-----------------------|---|-------------------------|-----------------------|
| | | | CBTC-SA diff. / shad. | CBTC-RD diff. / shad. | | CBTC-SA diff. / shad. | CBTC-RD diff. / shad. |
|  Conference | SAH | 10.5 / 7.8 | 0.88 / 0.82 | 0.86 / 0.79 | 7.1 / 5.2 | 0.88 / 0.84 | 0.82 / 0.77 |
| | AAC | 8.5 / 6.0 | 0.87 / 0.86 | 0.95 / 0.99 | 6.1 / 4.3 | 0.75 / 0.71 | 0.75 / 0.69 |
| | SBVH | 9.7 / 7.6 | 0.86 / 0.84 | 0.86 / 0.81 | 6.6 / 5.3 | 0.76 / 0.70 | 0.82 / 0.78 |
|  Crytek-Sponza | SAH | 26.0 / 22.5 | 0.99 / 0.80 | 0.90 / 0.76 | 17.6 / 15.3 | 0.90 / 0.65 | 0.81 / 0.68 |
| | AAC | 22.7 / 16.5 | 0.87 / 0.79 | 0.84 / 0.76 | 14.2 / 10.7 | 0.87 / 0.85 | 0.81 / 0.74 |
| | SBVH | 18.3 / 14.3 | 0.99 / 0.77 | 0.90 / 0.71 | 11.9 / 9.5 | 0.95 / 0.69 | 0.87 / 0.77 |
|  Soda: inside | SAH | 14.4 / 12.2 | 0.99 / 0.91 | 0.82 / 0.76 | 9.8 / 8.5 | 0.93 / 0.77 | 0.65 / 0.64 |
| | AAC | 18.2 / 16.6 | 0.79 / 0.65 | 0.84 / 0.76 | 12.0 / 11.1 | 1.35 / 1.03 | 0.82 / 0.75 |
| | SBVH | 13.3 / 12.9 | 0.97 / 0.86 | 0.81 / 0.70 | 8.9 / 8.9 | 0.93 / 0.93 | 0.68 / 0.64 |
|  Soda: outside | SAH | 8.3 / 8.3 | 0.93 / 0.88 | 0.81 / 0.72 | 5.7 / 5.7 | 0.87 / 0.86 | 0.72 / 0.68 |
| | AAC | 9.0 / 9.2 | 0.96 / 1.02 | 0.93 / 0.91 | 6.9 / 7.1 | 0.79 / 0.78 | 0.70 / 0.67 |
| | SBVH | 6.6 / 7.0 | 1.02 / 1.01 | 0.87 / 0.81 | 4.5 / 4.8 | 0.95 / 0.98 | 0.79 / 0.77 |
|  Arabje | SAH | 22.7 / 14.9 | 0.92 / 0.93 | 0.86 / 0.87 | 14.2 / 9.8 | 0.98 / 0.94 | 0.82 / 0.77 |
| | AAC | 18.7 / 12.6 | 0.92 / 0.99 | 0.91 / 0.93 | 12.0 / 8.1 | 0.95 / 0.98 | 0.88 / 0.86 |
| | SBVH | 12.8 / 10.6 | 0.99 / 1.00 | 0.88 / 0.88 | 8.8 / 7.3 | 0.94 / 0.95 | 0.78 / 0.82 |
|  Babylonian | SAH | 9.3 / 11.1 | 0.89 / 0.89 | 0.83 / 0.77 | 6.0 / 7.1 | 0.91 / 0.92 | 0.80 / 0.77 |
| | AAC | 10.2 / 11.8 | 0.96 / 0.95 | 0.85 / 0.80 | 6.7 / 7.8 | 0.96 / 0.90 | 0.84 / 0.77 |
| | SBVH | 6.9 / 8.7 | 0.93 / 0.99 | 0.79 / 0.83 | 4.7 / 6.0 | 0.91 / 0.96 | 0.74 / 0.72 |
|  Train-Station | SAH | 9.8 / 9.5 | 0.96 / 1.03 | 0.84 / 0.84 | 6.8 / 6.6 | 0.89 / 0.92 | 0.77 / 0.76 |
| | AAC | 10.9 / 10.2 | 0.94 / 0.88 | 0.85 / 0.85 | 7.3 / 6.8 | 0.90 / 0.80 | 0.76 / 0.76 |
| | SBVH | 8.9 / 8.7 | 0.96 / 0.93 | 0.82 / 0.83 | 6.0 / 6.0 | 0.95 / 0.91 | 0.77 / 0.76 |
|  Episode2 | SAH | 18.4 / 12.6 | 1.00 / 1.00 | 0.84 / 0.78 | 12.0 / 8.2 | 1.05 / 1.11 | 0.80 / 0.70 |
| | AAC | 18.5 / 12.6 | 0.98 / 0.98 | 0.86 / 0.79 | 13.2 / 9.1 | 0.94 / 0.96 | 0.70 / 0.61 |
| | SBVH | 17.3 / 11.1 | 1.03 / 1.04 | 0.83 / 0.79 | 11.6 / 7.6 | 0.99 / 1.00 | 0.73 / 0.65 |
|  Warehouse | SAH | 18.6 / 15.4 | 0.95 / 1.00 | 0.86 / 0.86 | 12.2 / 9.9 | 0.96 / 1.04 | 0.80 / 0.89 |
| | AAC | 19.3 / 16.2 | 0.98 / 0.99 | 0.83 / 0.86 | 13.0 / 11.6 | 0.88 / 0.86 | 0.74 / 0.68 |
| | SBVH | 14.8 / 14.3 | 1.00 / 1.23 | 0.89 / 1.00 | 10.1 / 9.7 | 0.98 / 1.01 | 0.81 / 0.82 |
|  San-Miguel | SAH | 35.4 / 18.7 | 0.99 / 0.71 | 0.92 / 0.79 | 23.7 / 12.7 | 0.96 / 0.87 | 0.88 / 0.72 |
| | AAC | 34.3 / 16.5 | 1.01 / 1.12 | 0.97 / 1.00 | 22.4 / 10.8 | 0.92 / 1.01 | 0.90 / 0.89 |
| | SBVH | 26.4 / 15.9 | 0.97 / 0.96 | 0.92 / 0.80 | 17.8 / 10.7 | 0.98 / 0.93 | 0.89 / 0.85 |
| Average 10 | SAH | | 0.95 / 0.90 | 0.85 / 0.79 | | 0.93 / 0.89 | 0.79 / 0.74 |
| | AAC | | 0.93 / 0.93 | 0.88 / 0.85 | | 0.93 / 0.89 | 0.79 / 0.74 |
| | SBVH | | 0.97 / 0.96 | 0.86 / 0.82 | | 0.93 / 0.91 | 0.79 / 0.76 |

Table 8: Numbers of SIMD ray-box tests (diffuse ray / shadow ray) by directly collapsing, and relative ratios with CBTC-SA and CBTC-RD. Experiments are based on both quad- and oct-BVH.

ray and 35% on shadow ray on Oct-BVH, by our testing). Our algorithm, however, is able to generate relatively high-quality Oct-BVH, which provides a chance to trade off between memory-bandwidth (less than 10% extra ray-box tests compared with non-optimized binary SAH BVH) and parallelism (8-way vs. 4-way).

| Scene | Relative ray-box tests | | Relative runtime RDTC |
|--------------|------------------------|-------------|-----------------------|
| | SATC | RDTC | |
| Darb.-Sponza | 0.87 / 1.00 | 0.79 / 0.67 | 0.80 / 0.76 |
| Fairy | 0.92 / 0.96 | 0.82 / 0.78 | 0.86 / 0.77 |
| Buddha | 0.98 / 0.89 | 0.93 / 0.91 | 0.94 / 0.96 |
| Powerplant | 0.88 / 0.84 | 0.84 / 0.78 | 0.86 / 0.82 |
| Hairball | 0.91 / 0.88 | 0.92 / 0.87 | 0.97 / 0.94 |

Table 9: Relative ratios of numbers of ray-box tests for diffuse/shadow rays, for the rest 5 scenes. Contractions are based on an initial BVHs generated by a top-down full-sweep SAH. Full results are shown in supplemental material.

5. Detail Results for SATC and RDTC

| Scene | initial BVH type | # box testing non-opt BVH diff / shad | relative performance | | relative runtime RDTC diff / shad | BVH imbalance | | reconstructed nodes | |
|---|------------------|---------------------------------------|----------------------|------------------|-----------------------------------|---------------|------|---------------------|---------|
| | | | SATC diff / shad | RDTC diff / shad | | SATC | RDTC | number | percent |
|  | SAH | 78.0 / 44.5 | 0.87 / 1.00 | 0.79 / 0.67 | 0.80 / 0.76 | 0.33 | 0.43 | 5.5K | 12.9% |
| | AAC | 78.9 / 44.3 | 0.88 / 0.64 | 0.81 / 0.60 | 0.79 / 0.66 | 0.22 | 0.39 | 4.7K | 13.2% |
| | SBVH | 55.6 / 29.6 | 0.96 / 1.01 | 0.86 / 0.83 | 0.89 / 0.87 | 0.24 | 0.40 | 6.2K | 10.1% |
|  | SAH | 56.5 / 28.4 | 0.92 / 0.96 | 0.82 / 0.78 | 0.86 / 0.77 | 0.27 | 0.39 | 9.0K | 8.0% |
| | AAC | 56.8 / 29.2 | 0.92 / 0.93 | 0.81 / 0.72 | 0.82 / 0.76 | 0.18 | 0.41 | 7.2K | 8.6% |
| | SBVH | 52.4 / 32.4 | 1.00 / 0.97 | 0.85 / 0.76 | 0.87 / 0.76 | 0.23 | 0.35 | 9.2K | 7.3% |
|  | SAH | 33.9 / 46.7 | 0.98 / 0.89 | 0.93 / 0.91 | 0.94 / 0.96 | 0.12 | 0.29 | 2.1K | 0.3% |
| | AAC | 40.0 / 49.2 | 0.94 / 0.96 | 0.91 / 0.86 | 0.91 / 0.90 | 0.15 | 0.27 | 2.5K | 0.4% |
| | SBVH | 33.0 / 44.1 | 0.98 / 0.98 | 0.92 / 0.89 | 1.01 / 0.98 | 0.13 | 0.30 | 2.0K | 0.2% |
|  | SAH | 73.0 / 53.2 | 0.88 / 0.84 | 0.84 / 0.78 | 0.86 / 0.82 | 0.30 | 0.38 | 5.9K | 3.0% |
| | AAC | 70.6 / 50.5 | 0.89 / 0.88 | 0.86 / 0.81 | 0.89 / 0.78 | 0.22 | 0.34 | 6.0K | 3.3% |
| | SBVH | 48.9 / 41.8 | 0.91 / 0.88 | 0.86 / 0.85 | 0.94 / 0.90 | 0.25 | 0.35 | 6.6K | 2.3% |
|  | SAH | 109.6 / 85.7 | 0.91 / 0.88 | 0.92 / 0.87 | 0.97 / 0.94 | 0.13 | 0.25 | 21.3K | 1.1% |
| | AAC | 136.2 / 97.2 | 0.86 / 0.92 | 0.84 / 0.88 | 0.81 / 0.94 | 0.15 | 0.25 | 21.3K | 1.2% |
| | SBVH | 102.8 / 80.5 | 0.91 / 0.90 | 0.92 / 0.88 | 0.98 / 0.95 | 0.14 | 0.26 | 20.0K | 0.6% |

Table 10: Detail experimental results for different scenes with various initial BVHs. Results for numbers of ray-box tests for non-optimized BVH, relative performance ratios for both SATC and RDTC, relative runtime ratios for actual wall-clock time for ray-primitive testing for RDTC, BVH imbalance descriptors for both SATC and RDTC, and reconstructed BVH nodes for RDTC.

6. The Relationship between Imbalance Descriptor and Relative Performance

To further demonstrate the inefficiency in BVH traversal, for a set of nodes S , we define the following “imbalance descriptor” (Imb) to measure the difference in probability for traversing the subtrees:

$$Imb(S) = \frac{\sum_{s \in S} visitCount(s) |\alpha_{N.left} - \alpha_{N.right}|}{\sum_{s \in S} visitCount(s)} \quad (2)$$

where α_N is measured by different parameters. Here the argument for Imb is all the nodes in a BVH, and the value range is between 0 and 1. This function predicts the acceleration by our method reasonably well, and we show the linear regressions between them here.

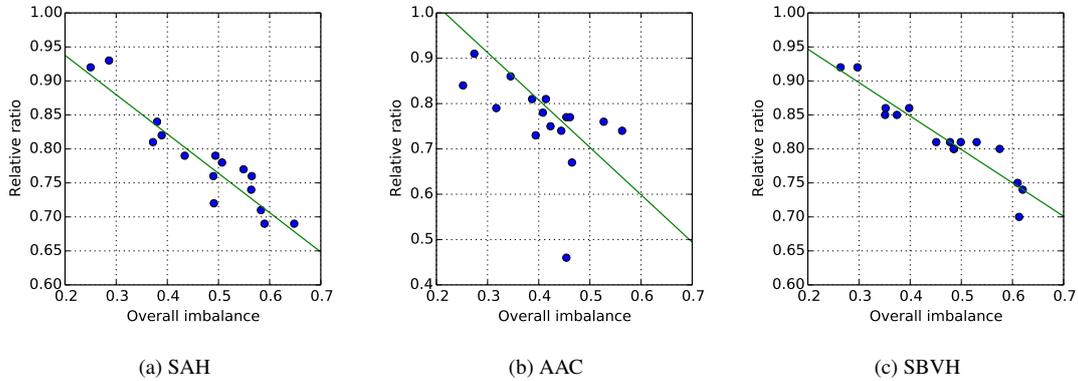


Figure 2: The linear regressions between imbalance descriptor and actual performance from all 15 scenes with RDTC, using the data shown in Table 1 in the paper and Table 1 in this document.

The regression function for SAH is $y = -0.5786x + 1.0534$, correlation: -0.8922 .

The regression function for AAC is $y = -1.0455x + 1.2261$, correlation: -0.7137 .

The regression function for SBVH is $y = -0.4916x + 1.0450$, correlation: -0.9438 .

Only one significant outlier is shown in the figures (SODA: INSIDE by AAC), which is caused by the special structure of the

model. For the BVHs built in top-down approaches like SAH and SBVH, the function *Imb* can provide an accurate estimation of the improvement by our method.

7. Detail Results for CBTC

| Scene | Initial BVH type type | # box testing non-optimized diff. / shad. | Quad-BVH relative ratio | | # box testing non-optimized diff. / shad. | Oct-BVH relative ratio | |
|---|-----------------------|---|-------------------------|-----------------------|---|------------------------|-----------------------|
| | | | CBTC-SA diff. / shad. | CBTC-RD diff. / shad. | | CBTC-SA diff. / shad. | CBTC-RD diff. / shad. |
|  | SAH | 19.2 / 11.9 | 0.96 / 0.84 | 0.93 / 0.78 | 13.1 / 8.2 | 0.92 / 0.82 | 0.85 / 0.72 |
| | AAC | 19.5 / 12.1 | 1.00 / 1.03 | 0.95 / 0.96 | 13.3 / 8.3 | 0.98 / 1.04 | 0.85 / 0.88 |
| | SBVH | 14.3 / 8.5 | 1.03 / 1.11 | 0.93 / 0.78 | 9.6 / 5.7 | 1.00 / 1.07 | 0.89 / 0.93 |
|  | SAH | 14.3 / 7.3 | 1.01 / 1.08 | 0.93 / 0.90 | 9.7 / 5.1 | 1.00 / 1.08 | 0.84 / 0.76 |
| | AAC | 15.1 / 7.6 | 1.01 / 1.09 | 0.92 / 0.89 | 10.2 / 5.5 | 0.94 / 0.86 | 0.86 / 0.75 |
| | SBVH | 12.9 / 8.3 | 1.05 / 1.06 | 0.93 / 0.80 | 8.9 / 5.4 | 1.04 / 1.08 | 0.90 / 0.95 |
|  | SAH | 8.8 / 13.3 | 1.00 / 1.03 | 0.95 / 0.99 | 6.0 / 9.0 | 0.99 / 1.01 | 0.94 / 1.01 |
| | AAC | 11.2 / 15.1 | 0.99 / 1.03 | 0.97 / 0.97 | 7.5 / 10.2 | 0.98 / 0.98 | 1.01 / 0.97 |
| | SBVH | 8.5 / 12.6 | 1.00 / 1.04 | 0.96 / 1.00 | 5.8 / 8.6 | 1.00 / 1.01 | 0.94 / 0.95 |
|  | SAH | 27.6 / 22.4 | 0.98 / 0.97 | 0.96 / 0.93 | 18.6 / 15.2 | 0.96 / 0.97 | 0.93 / 0.90 |
| | AAC | 36.9 / 26.2 | 0.95 / 1.00 | 0.94 / 0.96 | 23.3 / 17.6 | 0.98 / 1.00 | 0.96 / 0.94 |
| | SBVH | 25.9 / 21.2 | 0.98 / 0.99 | 0.96 / 0.97 | 17.4 / 14.3 | 0.96 / 0.97 | 0.94 / 0.93 |
|  | SAH | 18.3 / 14.4 | 0.95 / 1.01 | 0.93 / 0.93 | 12.3 / 9.9 | 0.91 / 0.93 | 0.86 / 0.83 |
| | AAC | 17.6 / 13.9 | 0.97 / 0.92 | 0.95 / 0.91 | 11.9 / 9.6 | 0.92 / 0.89 | 0.89 / 0.87 |
| | SBVH | 12.2 / 11.5 | 0.99 / 1.05 | 0.93 / 0.95 | 8.3 / 7.9 | 0.95 / 0.95 | 0.89 / 0.88 |
| Average 5 | SAH | | 0.98 / 0.99 | 0.94 / 0.91 | | 0.96 / 0.96 | 0.88 / 0.84 |
| | AAC | | 0.98 / 1.01 | 0.95 / 0.94 | | 0.96 / 0.95 | 0.91 / 0.88 |
| | SBVH | | 1.01 / 1.05 | 0.94 / 0.90 | | 0.99 / 1.02 | 0.91 / 0.93 |

Table 11: Numbers of SIMD ray-box tests (diffuse ray / shadow ray) by directly collapsing, and relative ratios with CBTC-SA and CBTC-RD. Experiments are based on both quad- and oct-BVH.

References

- [BWB08] BOULOS S., WALD I., BENTHIN C.: Adaptive Ray Packet Reordering. In *Proc. Interactive Ray Tracing* (2008). 1
- [DHK08] DAMMERTZ H., HANIK A., KELLER A.: Shallow bounding volume hierarchies for fast SIMD ray tracing of incoherent rays. In *Proc. Eurographics Conference on Rendering* (2008). 6
- [GHFB13] GU Y., HE Y., FATAHALIAN K., BLELLOCH G.: Efficient BVH construction via approximate agglomerative clustering. In *Proc. High-Performance Graphics* (2013). 1
- [GS87] GOLDSMITH J., SALMON J.: Automatic creation of object hierarchies for ray tracing. *IEEE Computer Graphics and Applications* 7, 5 (May 1987), 14–20. 1
- [Jak10] JAKOB W.: <http://mitsuba-renderer.org>. 1
- [PBD*10] PARKER S. G., BIGLER J., DIETRICH A., FRIEDRICH H., HOBEROCK J., LUEBKE D., MCALLISTER D., MCGUIRE M., MORLEY K., ROBISON A., ET AL.: Optix: a general purpose ray tracing engine. *ACM Transactions on Graphics (TOG)* 29, 4 (2010), 66. 1
- [SFD09] STICH M., FRIEDRICH H., DIETRICH A.: Spatial splits in bounding volume hierarchies. In *High-Performance Graphics* (2009). 1
- [Tsa09] TSAKOK J. A.: Faster incoherent rays: Multi-bvh ray stream tracing. In *Proc. High Performance Graphics* (2009). 1
- [WBB08] WALD I., BENTHIN C., BOULOS S.: Getting rid of packets - efficient SIMD single-ray traversal using multi-branching BVHs. In *Proc. Interactive Ray Tracing* (2008). 6
- [WWB*14] WALD I., WOOP S., BENTHIN C., JOHNSON G. S., ERNST M.: Embree: a kernel framework for efficient cpu ray tracing. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 143. 1