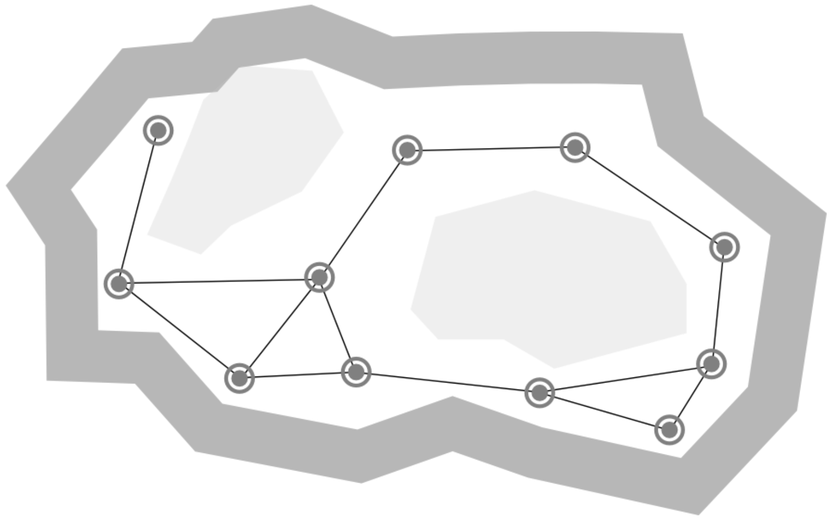
SIMDop:
SIMD Optimized Bounding Volume Hierarchies for Collision Detection

Toni Tan, René Weller, Gabriel Zachmann
University of Bremen, Germany
cgvr.informatik.uni-bremen.de

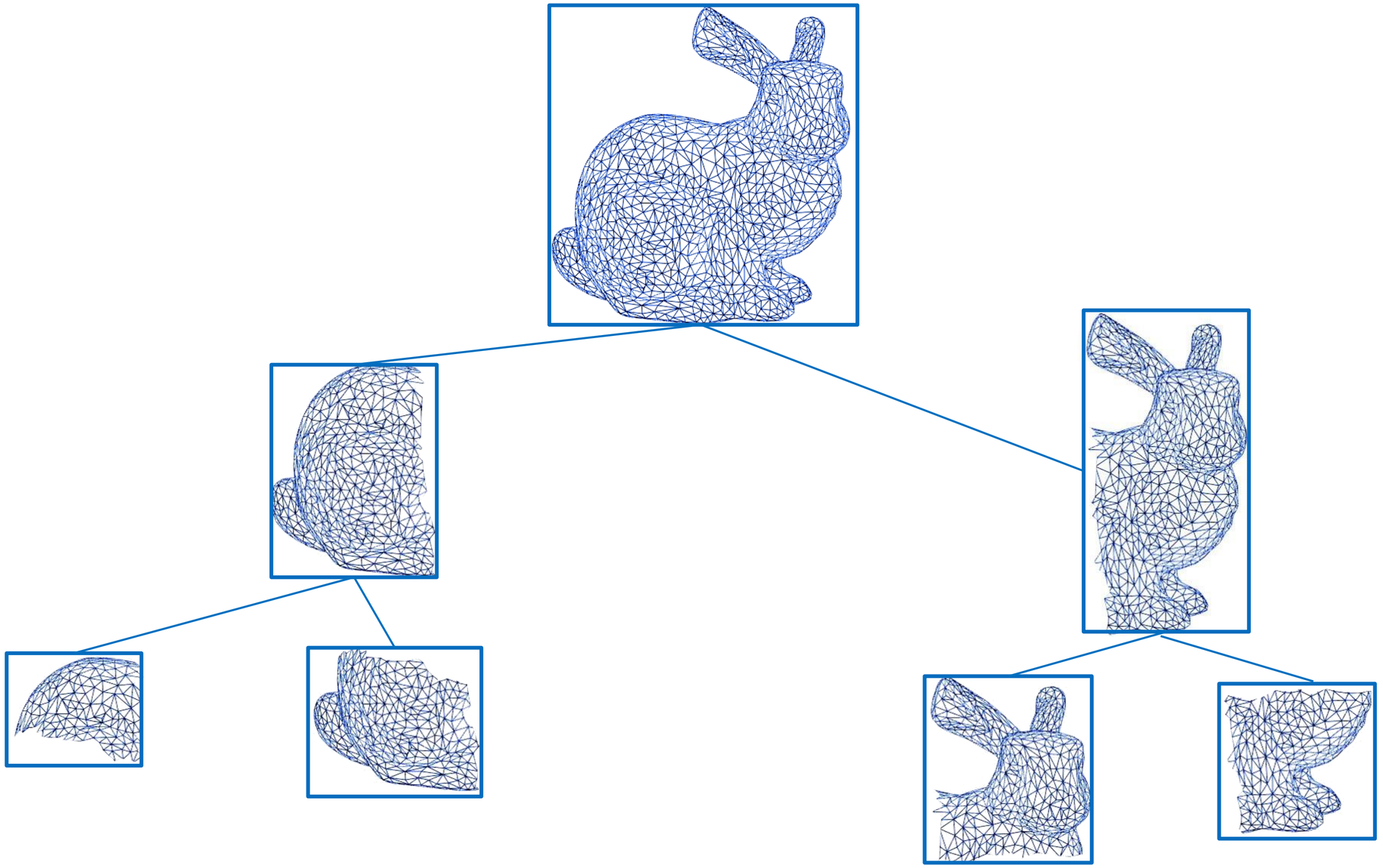
IROS, November 2019

Motivation

- Collision computation takes up to 90% in most sampling-based motion planning [Reggiani et al., 2002]
- Physically-based simulation in virtual environment [Damkjær & Erleben, 2009]

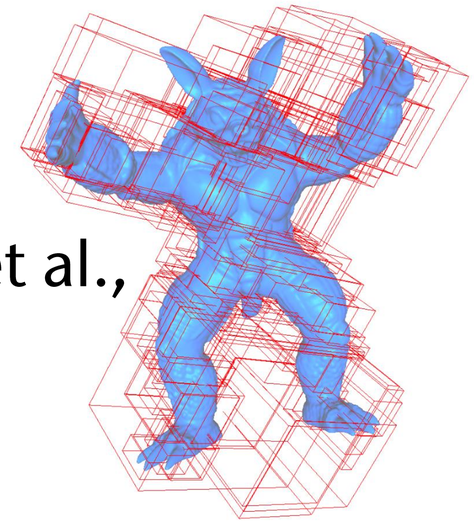


Bounding Volume Hierarchies (BVH)

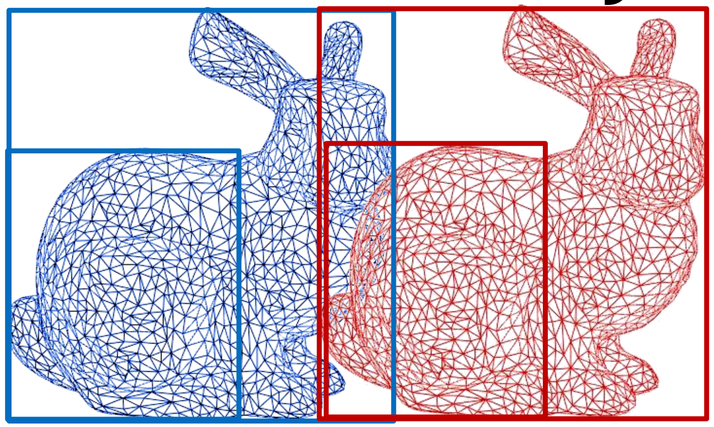


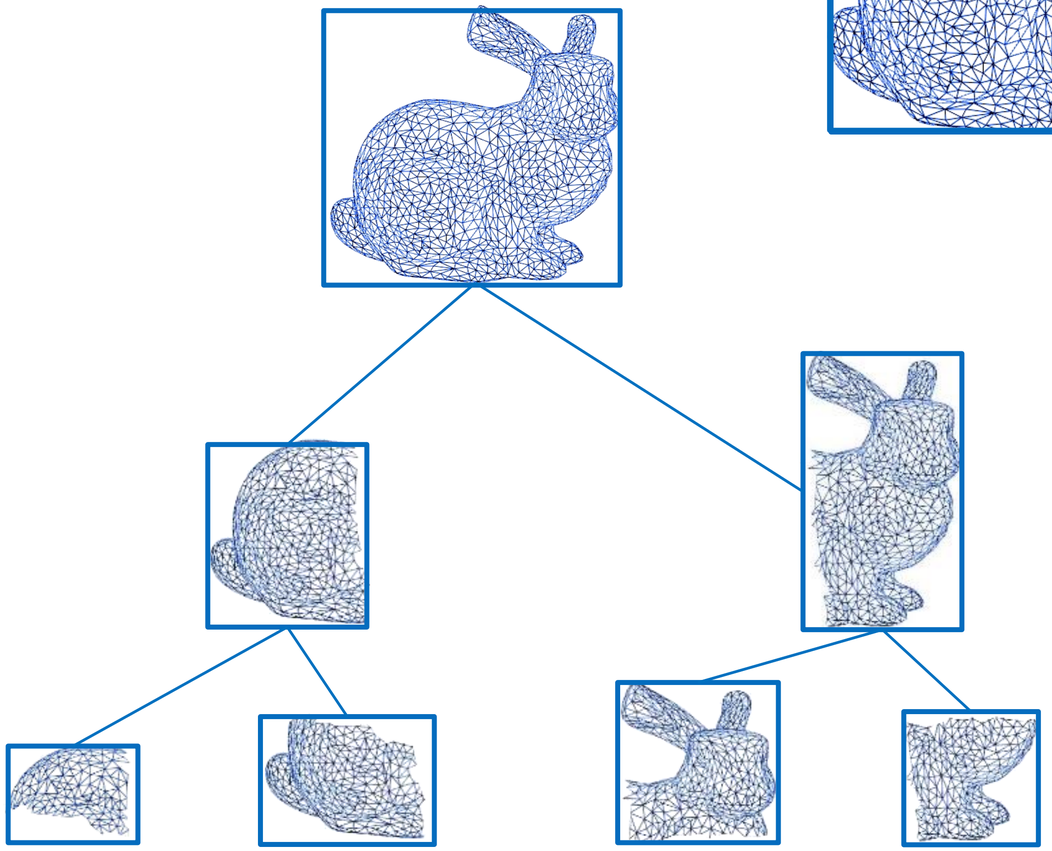
Previous Work

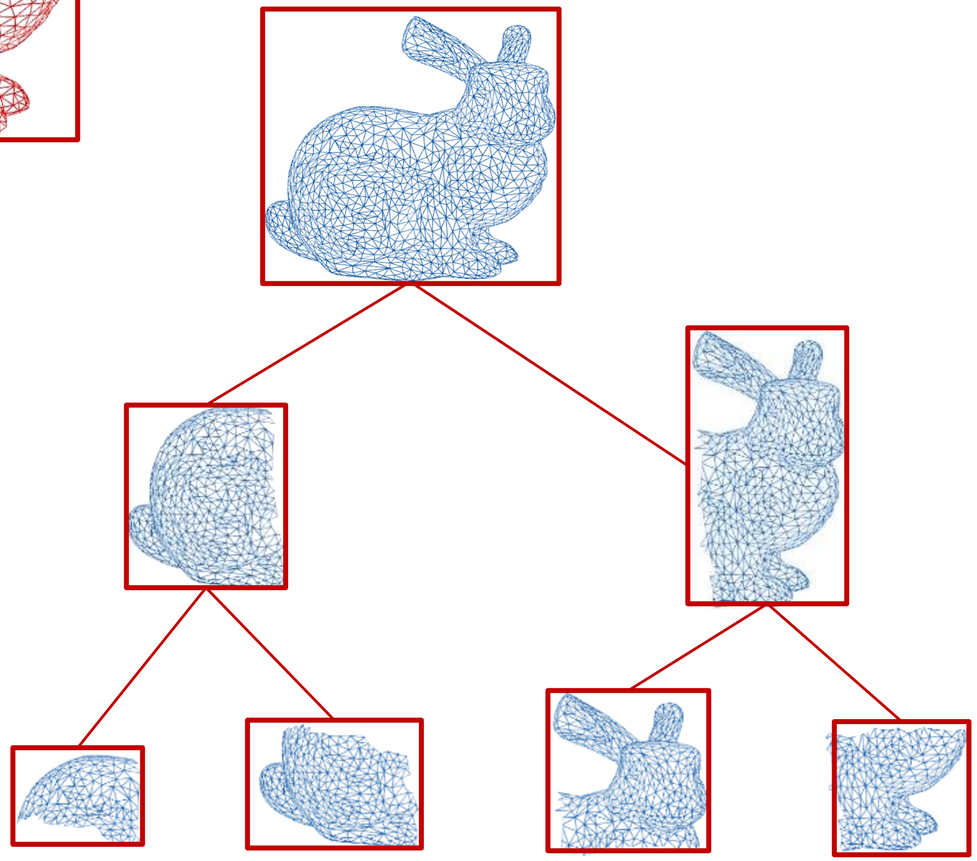
- BVH using various BVs:
  - Sphere [Hubbard, 1996], AABBs [Bergen, 1998] [Zachmann, 2002], k-DOP [Klosowski et al., 1998] [Zachmann, 1998], OBBs [Gottschalk et al., 1996], Convex hull trees [Ehmann & Ming, 2001]
- In sampling-based motion planning:
  - combination of several BVs (AABB, Sphere, OBB) [Ferguson et al., 2008]
  - AABB-based BVH (Schwesinger et al., 2015)



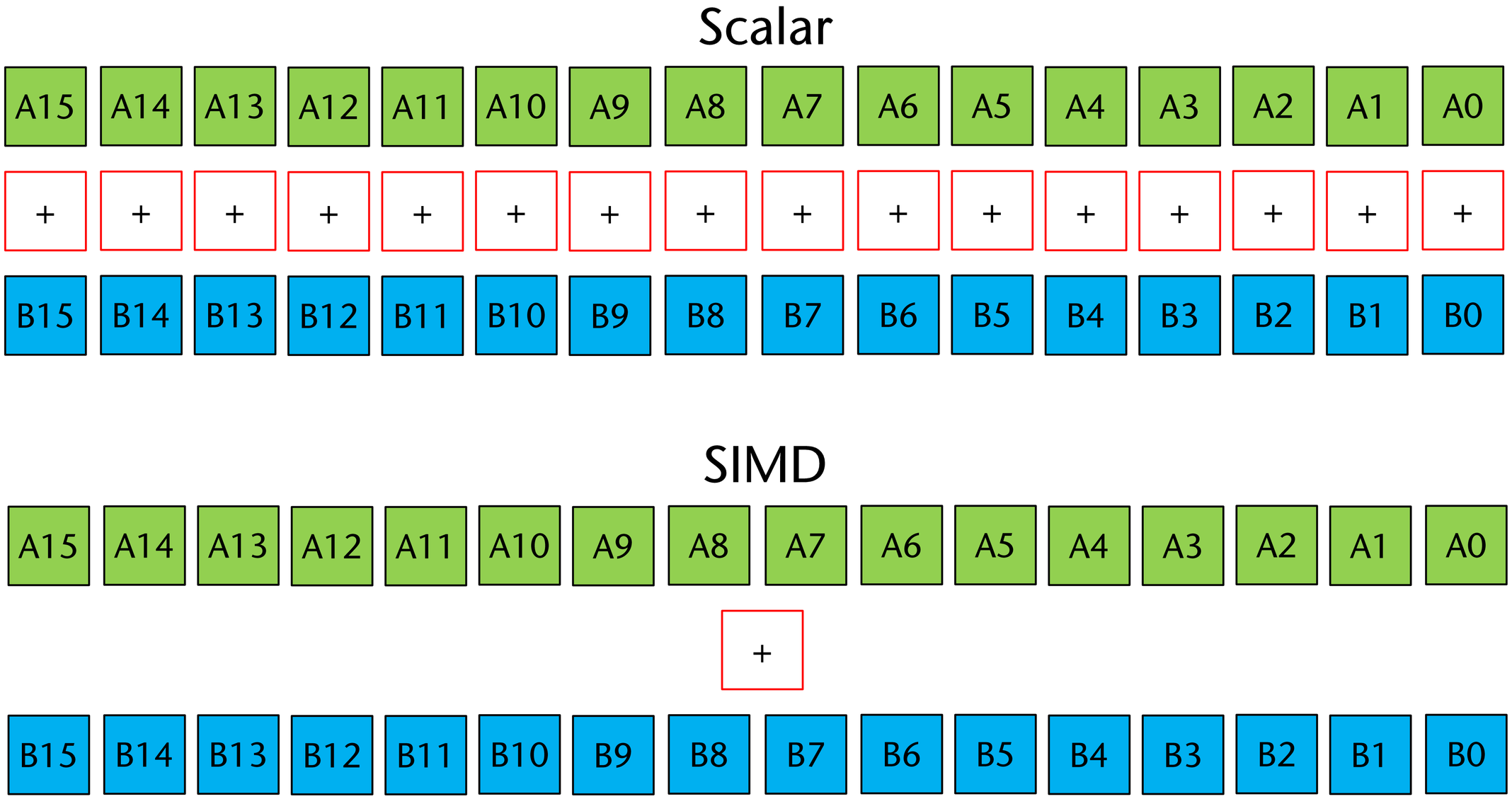
Bounding Volume Hierarchy Traversal





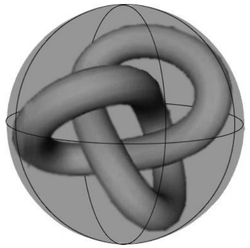


SIMD Instruction Sets (AVX512)

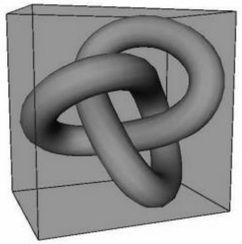


SIMD Optimization Methods

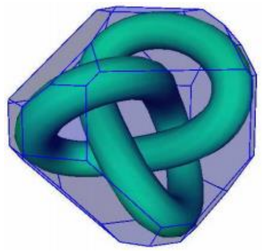
- Method 1: switch on compiler option */arch:AVX512*
- Method 2: optimize the BV overlap test manually



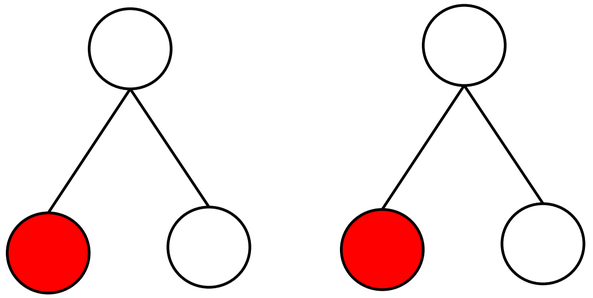
Sphere



AABB

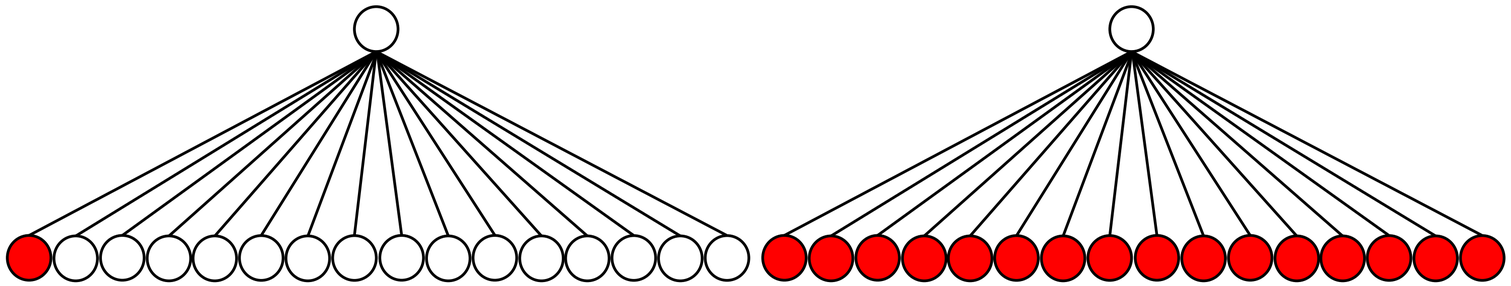


k-DOP

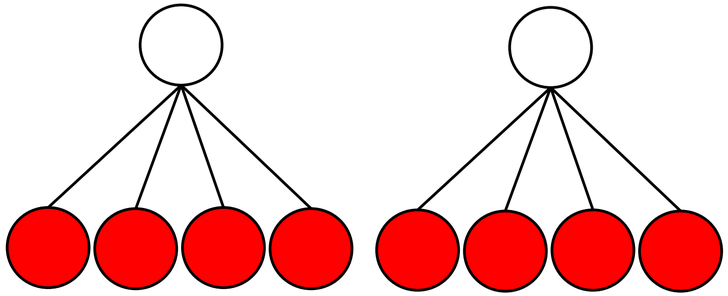


1 vs 1

- Method 3: redesign of the BVH topology



1 vs many



many vs many

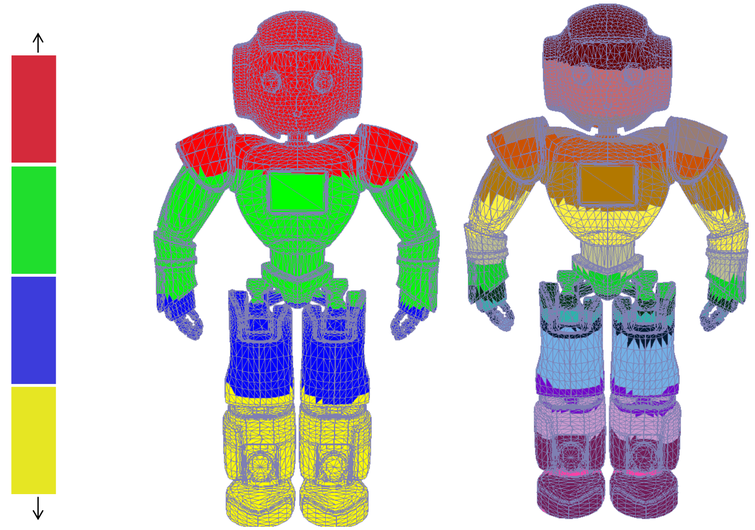
Our Contributions

1. **Novel BVHs with higher branching factor**

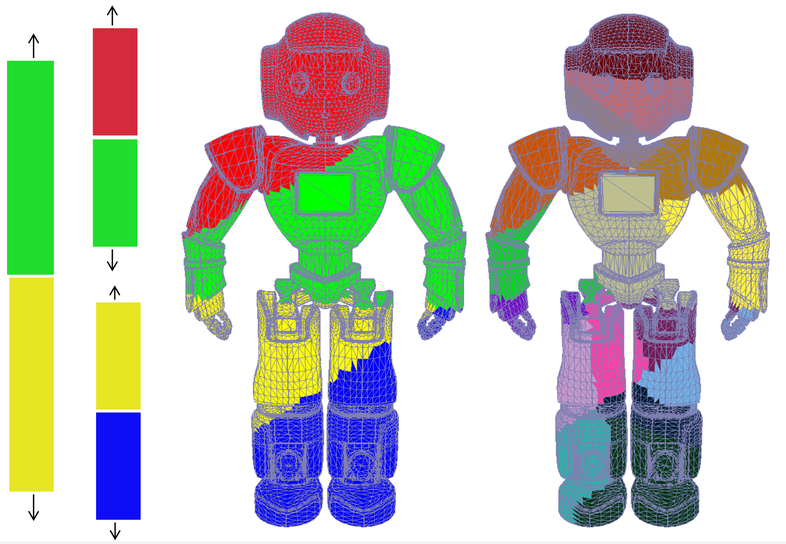
2. **SIMD optimized traversal algorithms**

BVH Construction Strategies

- Longest Axis Split
  - Construct the longest axis
    - [Dickerson et al., 2002] prove optimality for kd-tree (binary)
  - Sort polygons based on the axis
  - Partition into desired branching factors

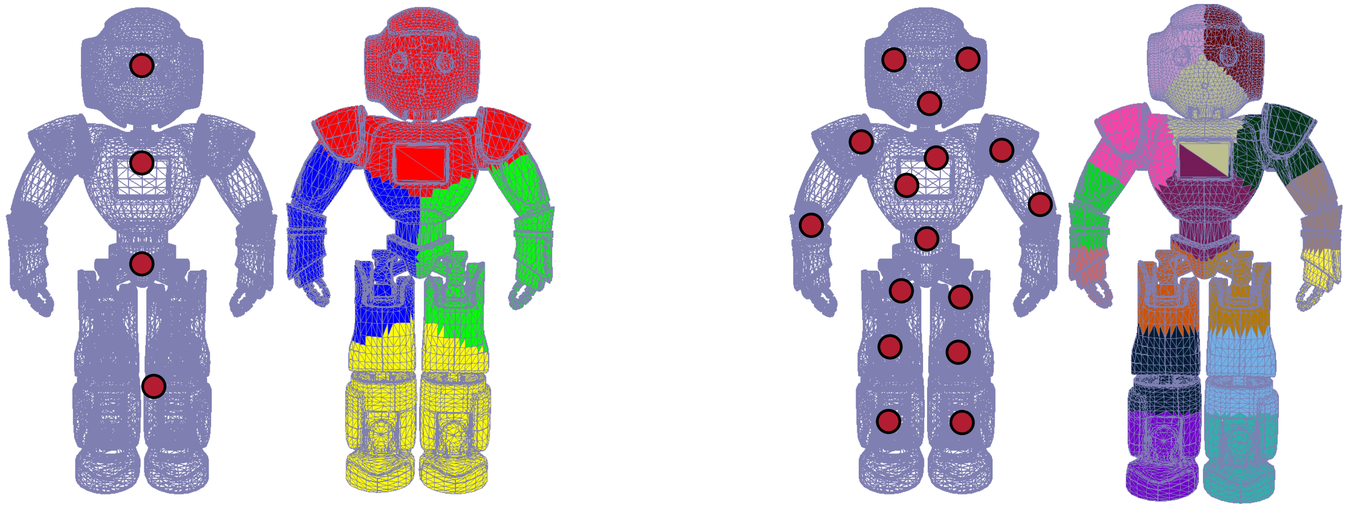


- Extended Longest Axis Split
  - Construct the longest axis
  - Sort polygons based on the axis, split into two
  - Construct longest axis again for both split parts recursively



BVH Construction Strategies

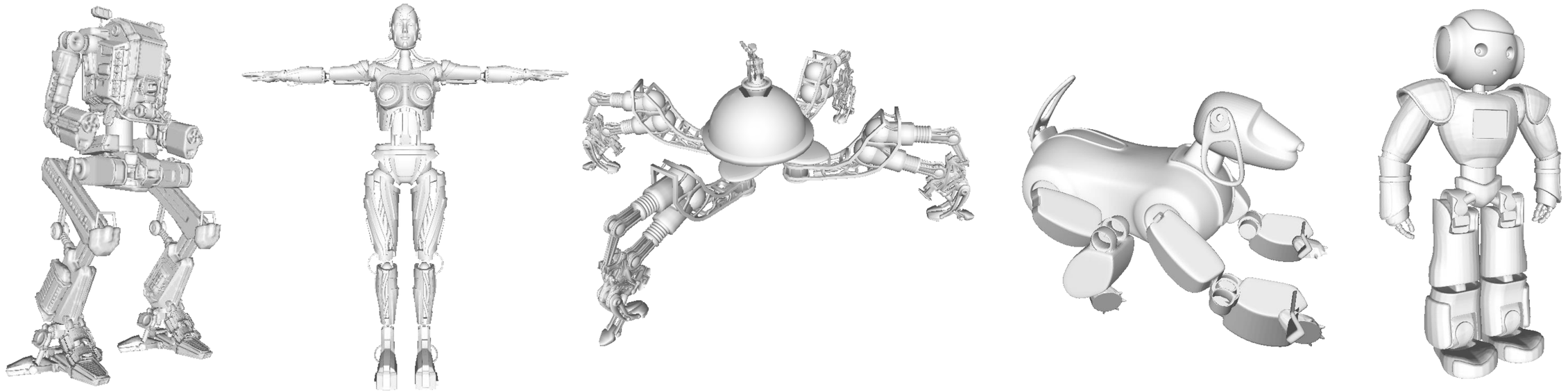
- Batch Neural Gas (BNG) Clustering [Weller et al., 2014]



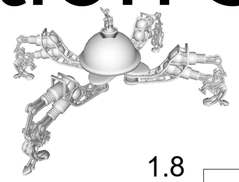
- Advantages:
  - Very robust behavior with respect to the initial cluster center position
  - Future-proof for further SIMD development

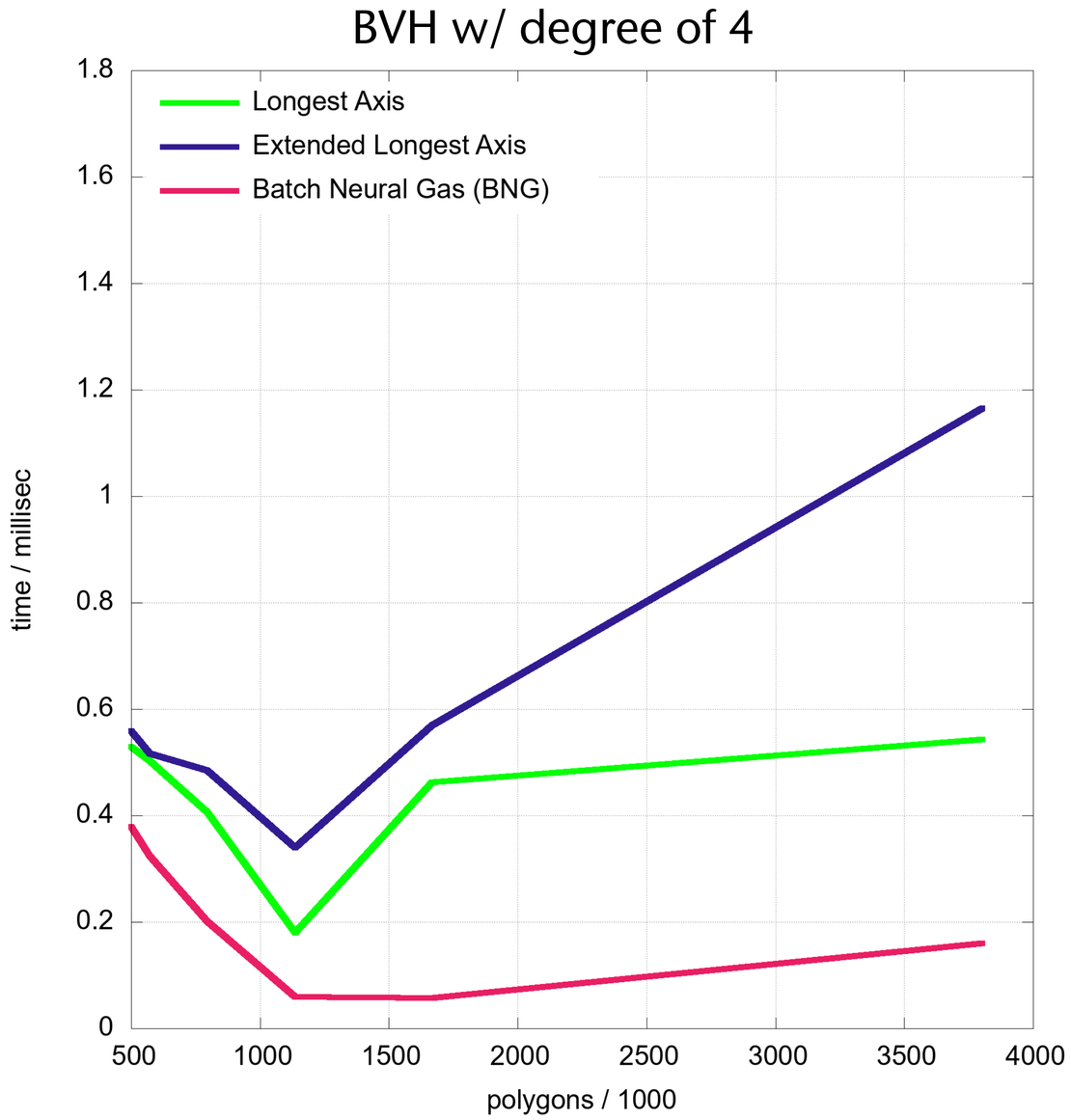
Benchmark

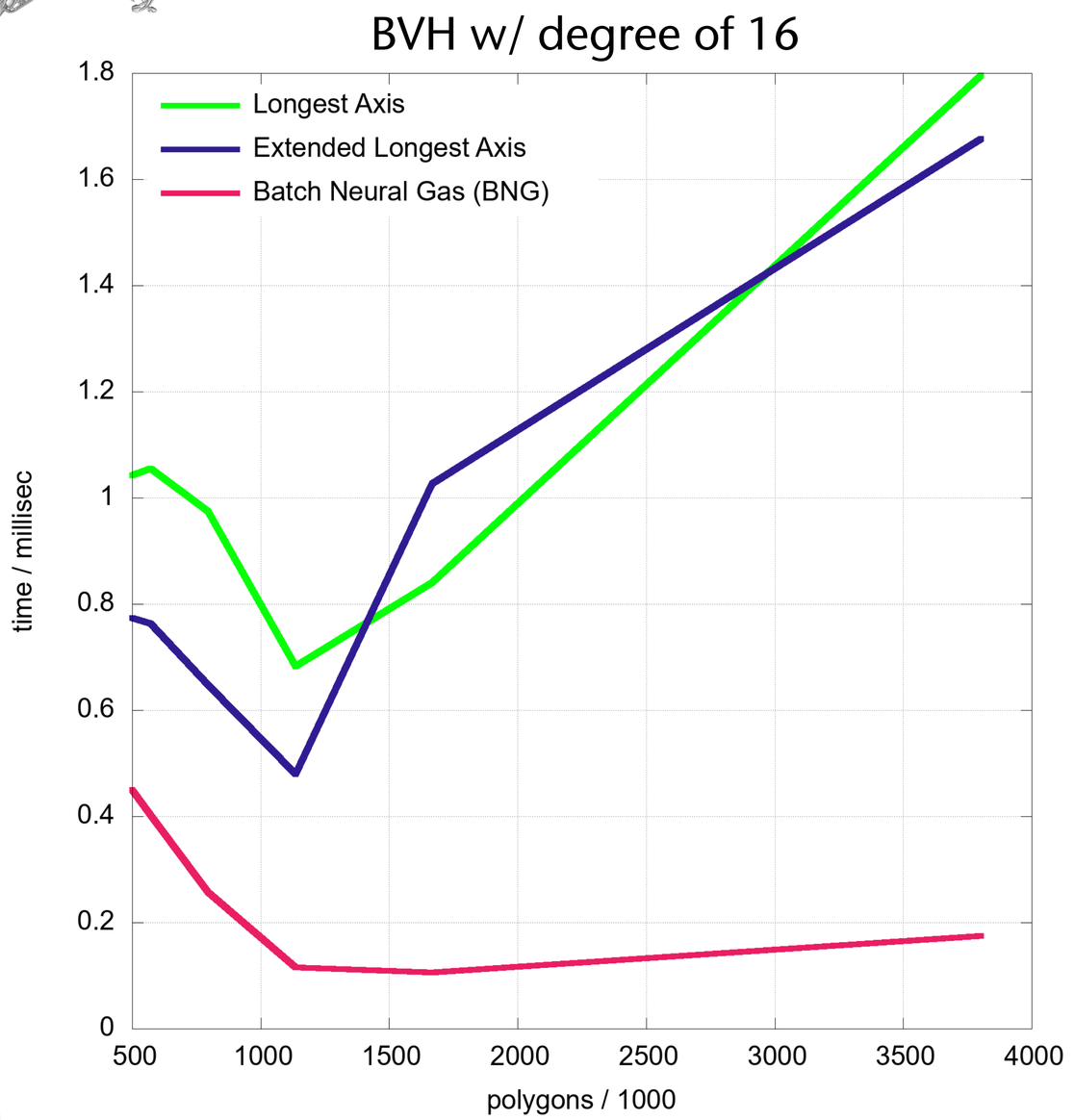
- Use benchmarking suite for collision detection proposed by [Trenkel et al., 2007]
- Use processor with AVX512 capability



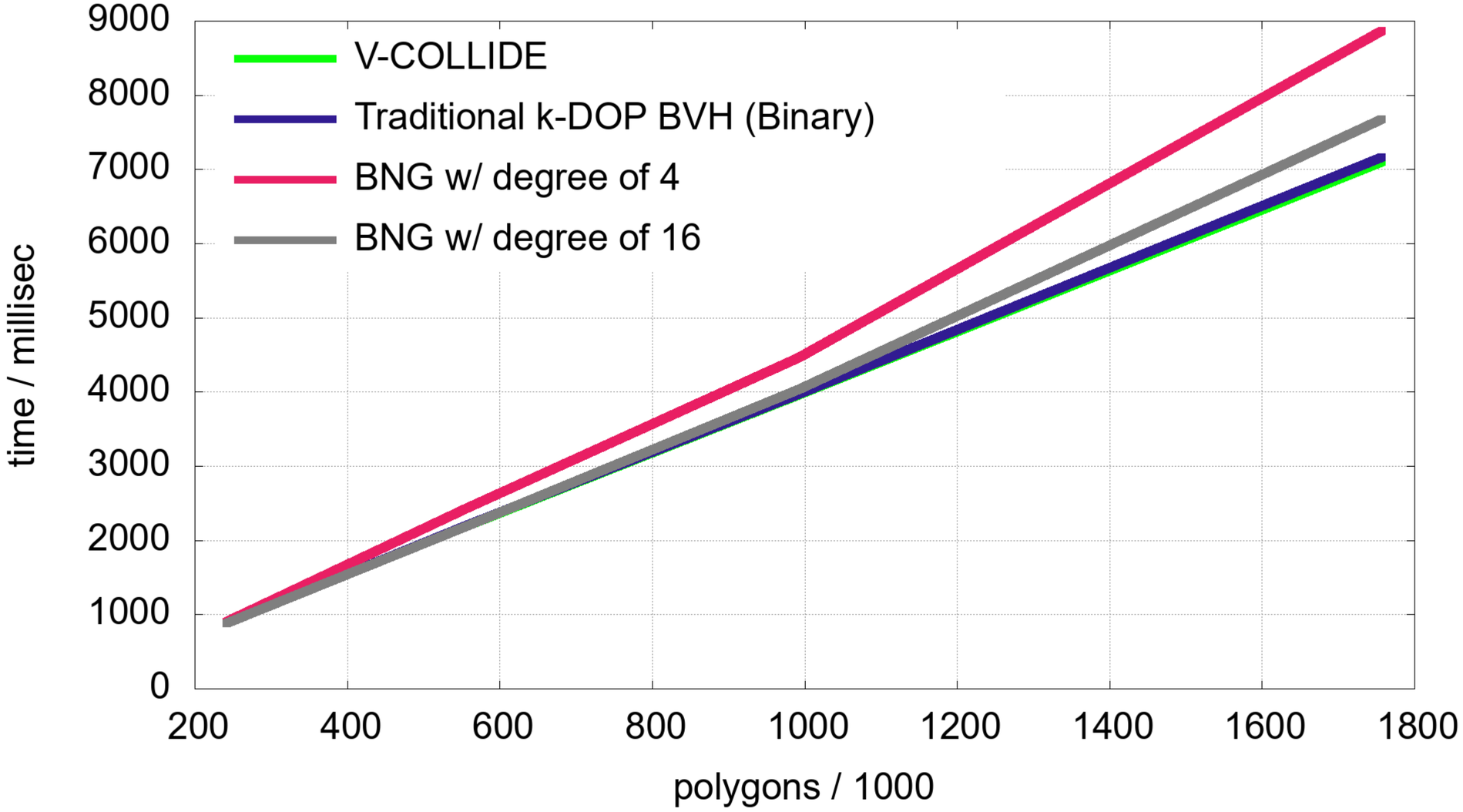
Results: BVH Construction Strategies (SISD)

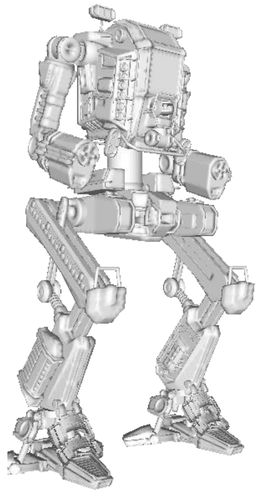




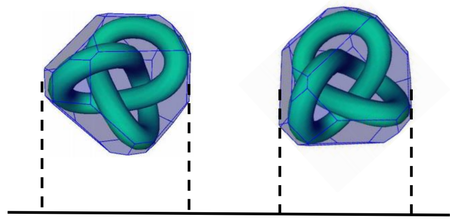


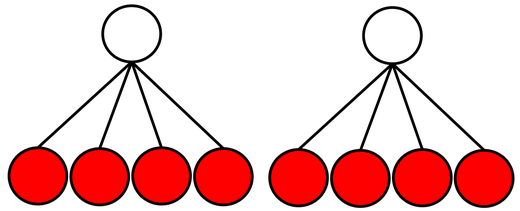
Result: BVH Construction Time





Simultaneous BVH Traversal Algorithm (4 vs 4)





```
for ( h = 0; h < degreeBVH; h++)
{
    for ( i = 0; i < degreeBVH; i++)
    {
        for ( j = 0; j < k / 2; j++ )
        {
            bool resL = a[j]  < -b[k / 2 + j];

            bool resH = -a[k/ 2 + j] >  b[j];

            if ( resL || resH )
                return false;
        }
        return true;
    }
}
```

k-DOP Intersection Test

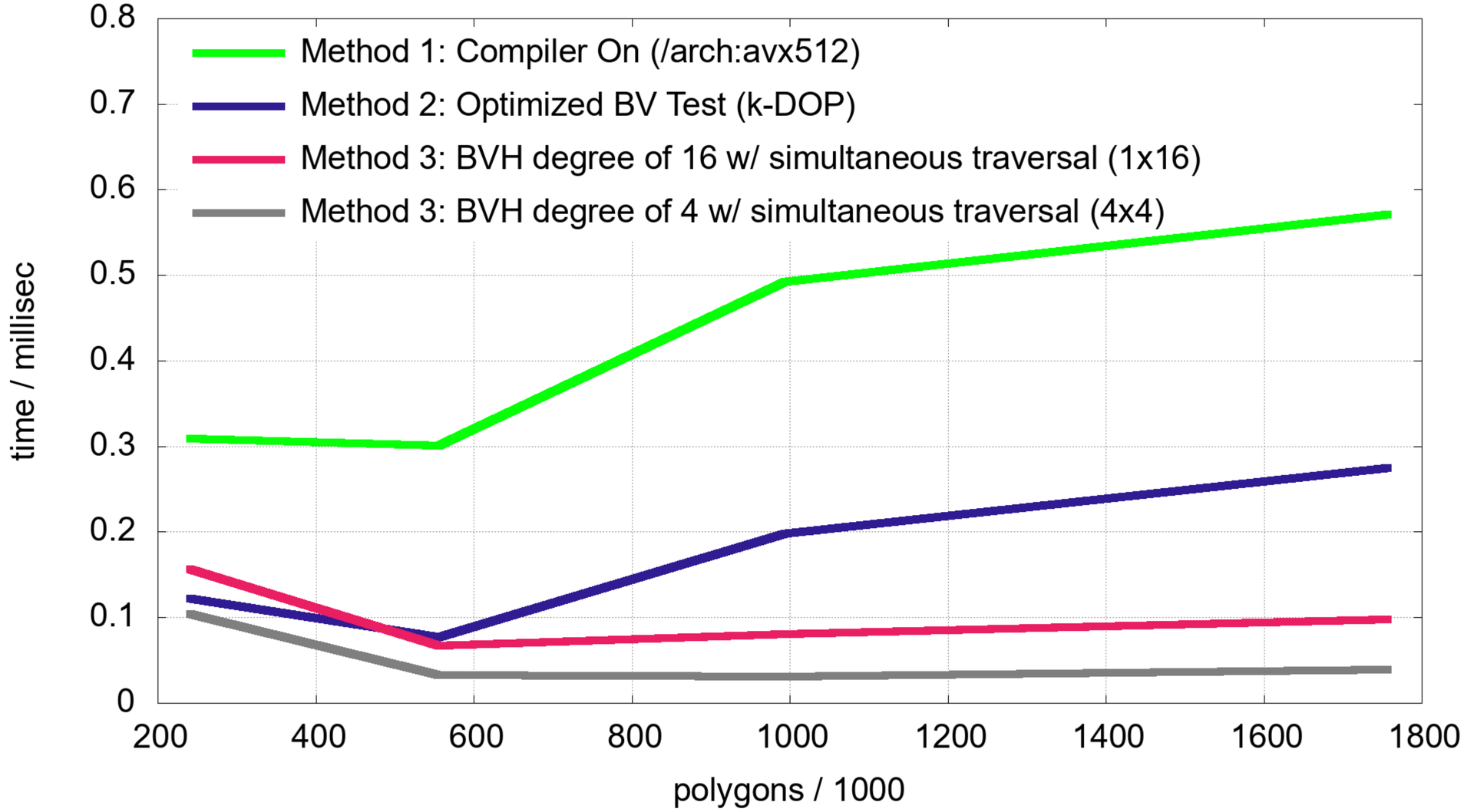
3 x 16 operations / orientation

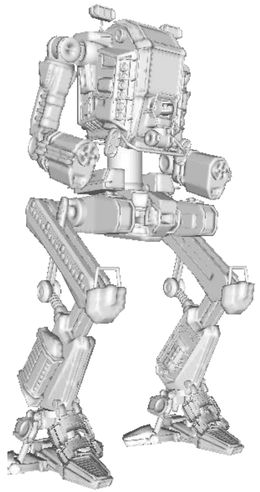
```
_mm512 endResult = _mm512_set1_ps( 1.0f );

for ( i = 0; i < k / 2; i++ )
{
    _mm512 resL = _mm512_cmp_ps( oriAL, oriBL, _CMP_LT_OS );

    _mm512 resH = _mm512_cmp_ps( oriAH, oriBH, _CMP_GT_OS );

    _mm512 tempRes = _mm512_kor( resL, resH );

    endResult = _mm512_kor ( endResult, tempRes );

    If ( endResult == 65535 )
        break;
}
return endResult;
```

parallel k-DOP Intersection Test

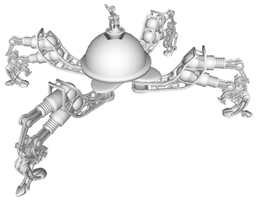
5 operations / orientation

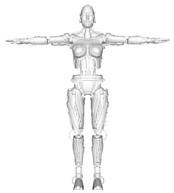
Result: SIMD Implementations

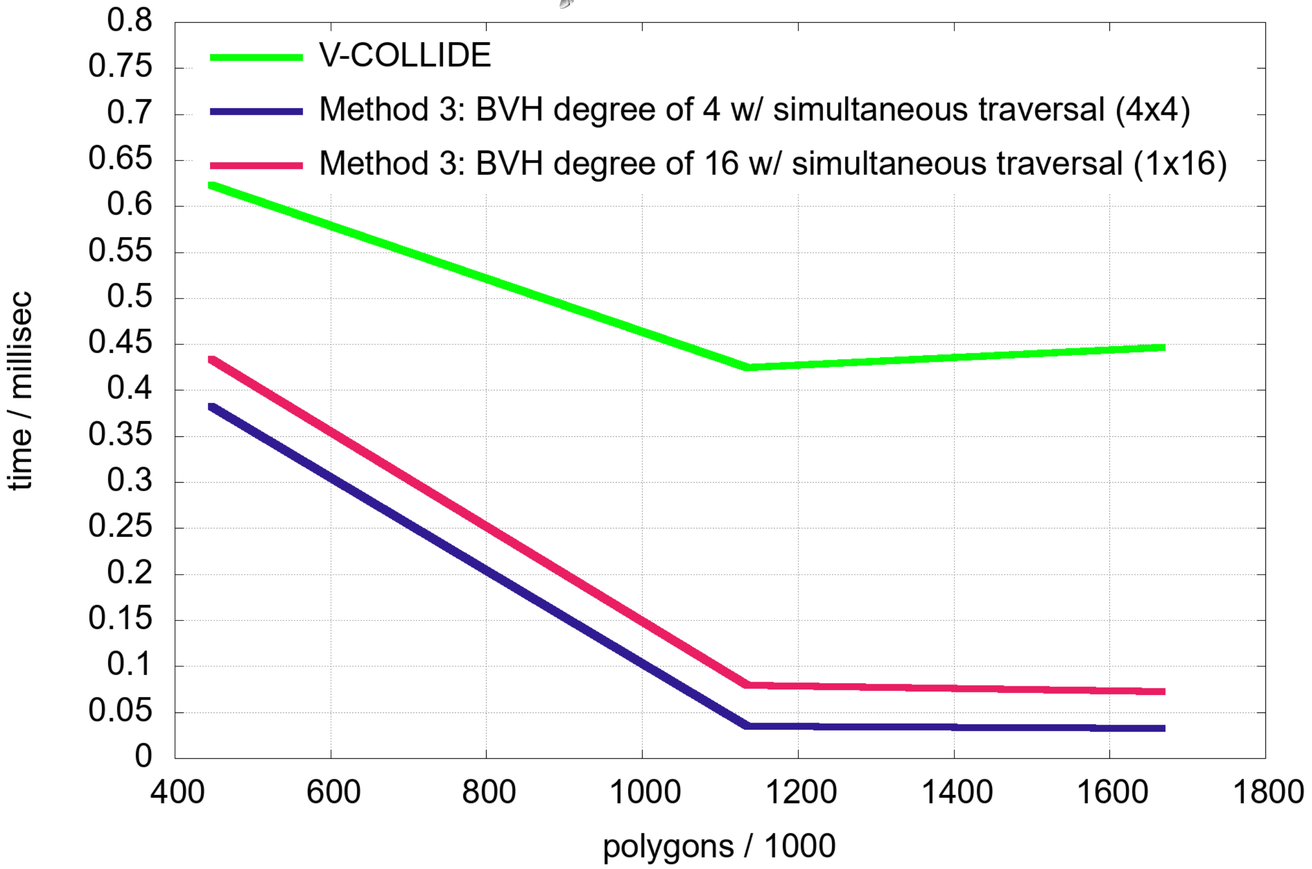


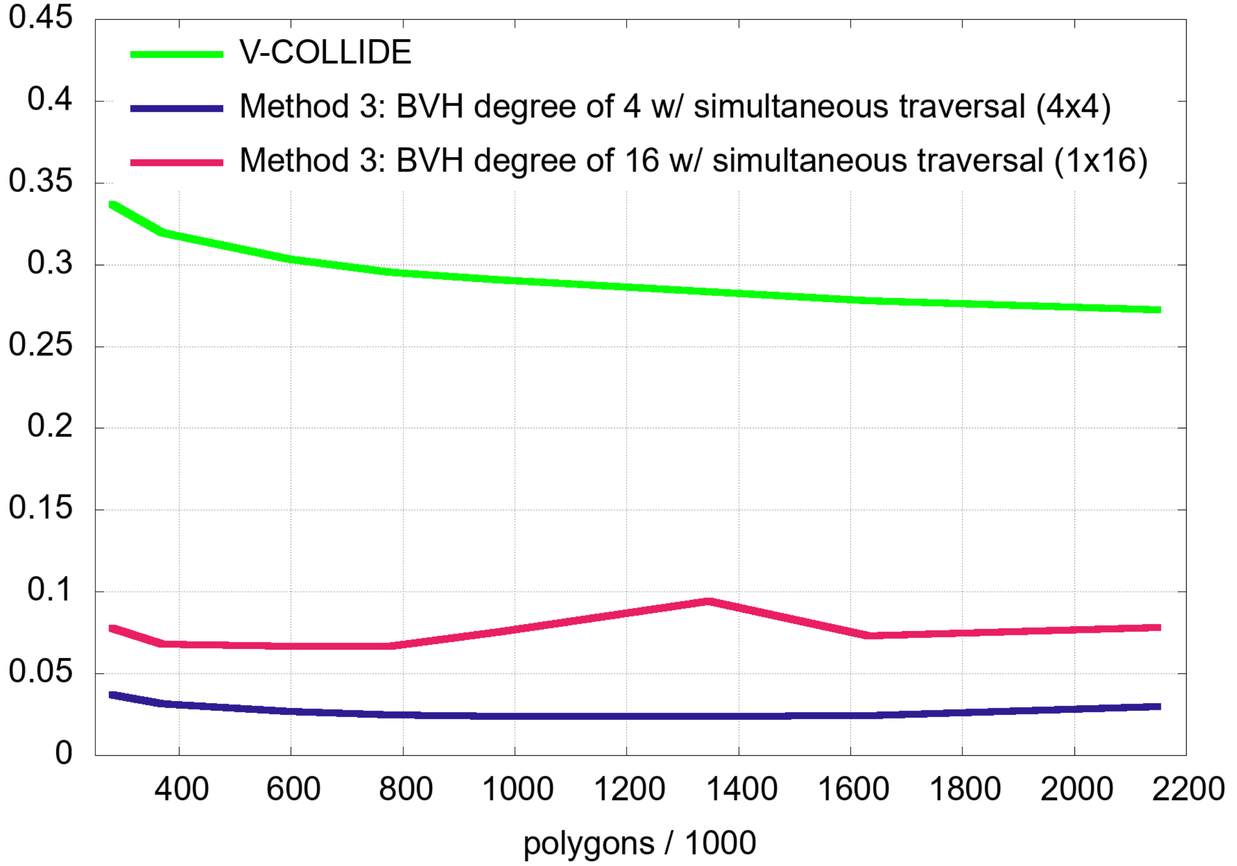


Results: Comparison with V-COLLIDE





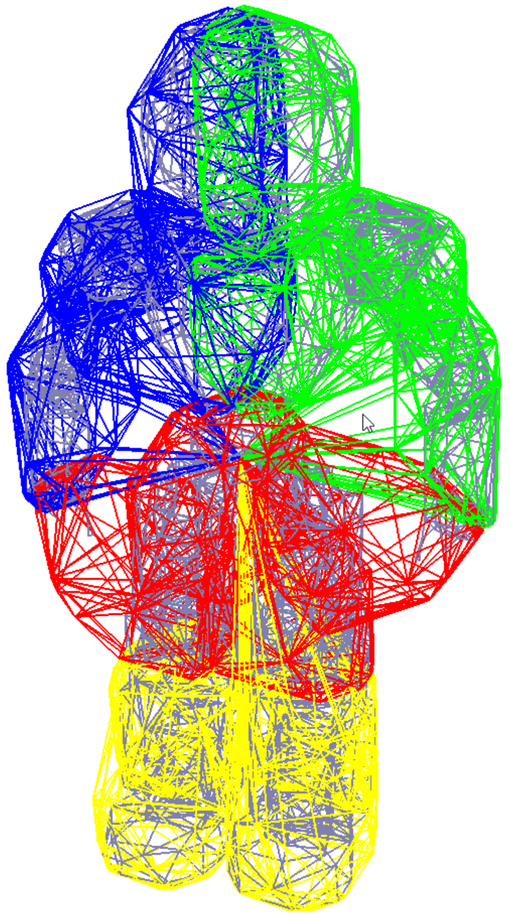




(An experimental comparative analysis has shown that V-COLLIDE outperforms other CD libraries like PQP [Reggiani et al., 2002])

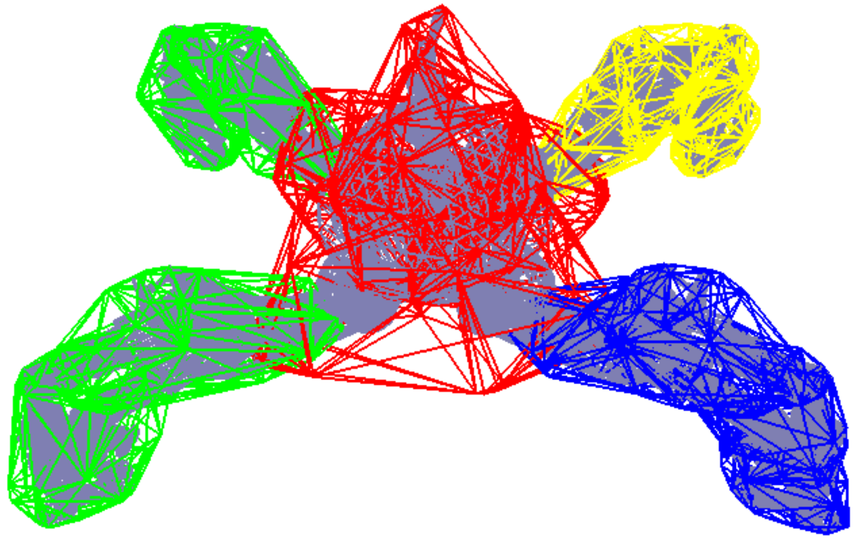
Conclusions

- Novel BVH with higher branching factors
- Novel heuristic for constructing BVH with arbitrary branching factor (BNG)
- Novel SIMD optimized traversal algorithm
- Our SIMD Optimized BVH outperforms traditional BVHs by an order of magnitude

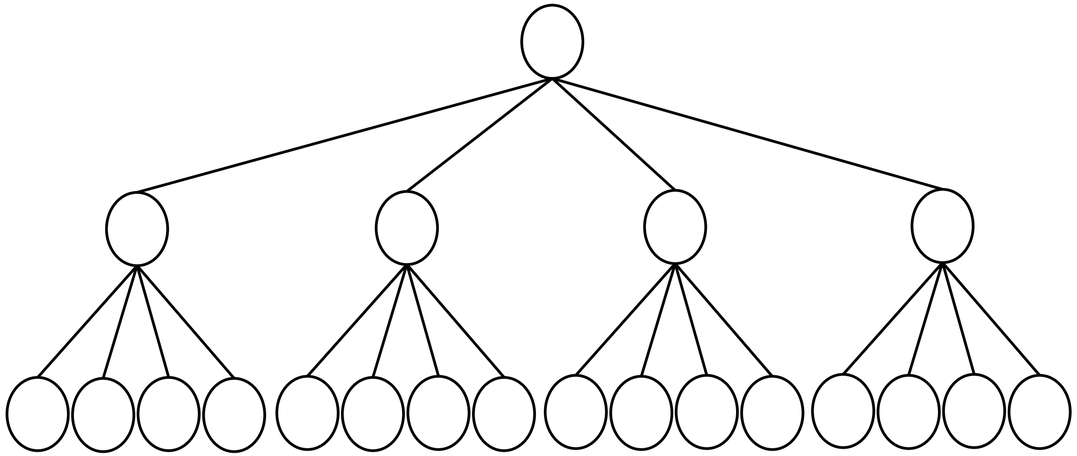


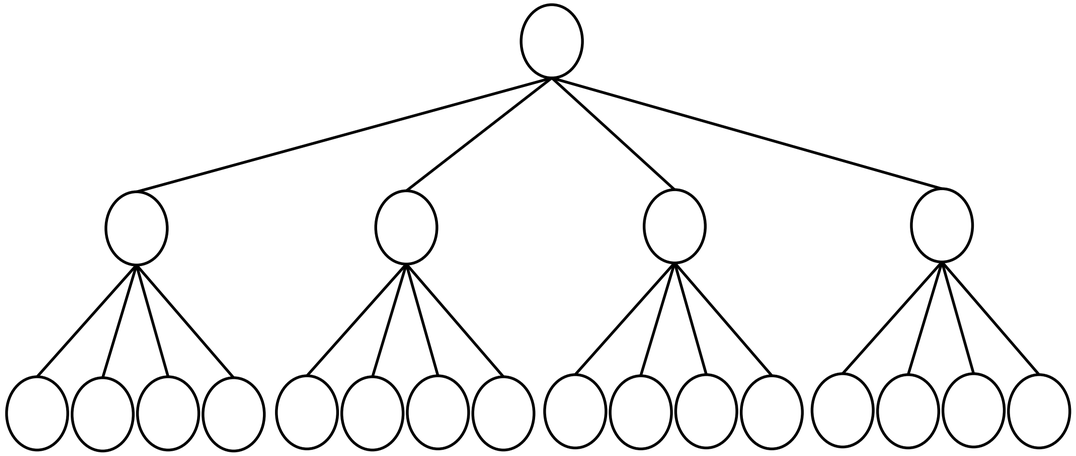
Future Works

- Explore other BV for SIMD traversal algorithms
- Influence of the number of orientations for k-DOP
- Include magnification control to BNG construction algorithm



Thank You!





Toni Tan, René Weller, Gabriel Zachmann
{toni, weller, zach}@cs.uni-bremen.de