

VDB

High-Resolution Sparse Volumes with Dynamic Topology

Outline

- Goal
- Data structure
- Applications
- Results

Goal

- Efficient storage of voxel data
 - Sparse data
 - Level set
- Efficient access
 - Random
 - Sequential
 - Stencil
- Store dynamic data in a dynamic structure

Characteristics according to paper

- Dynamic
- Memory efficient
- General topology
- Fast random and sequential data access
- Virtually infinite
- Efficient hierarchical algorithms
- Adaptive resolution
- Simple
- Configurable
- Out-of-core

Data Structure

- Hierarchy
 - Fixed height
 - Similar to bounding boxes
- Compression
 - Sparse data
 - Bit codes for active tiles
- Configurations
 - Height
 - Branching factor
 - Node size

Data Structure

x close

Data Structure

Root node
(unbounded)



Tile values with
active/inactive states

Internal Node 1



Active Mask
Child Mask
Tile values / Child pointers

Internal Node 2



Leaf Node



Active Mask
Voxels

Nodes

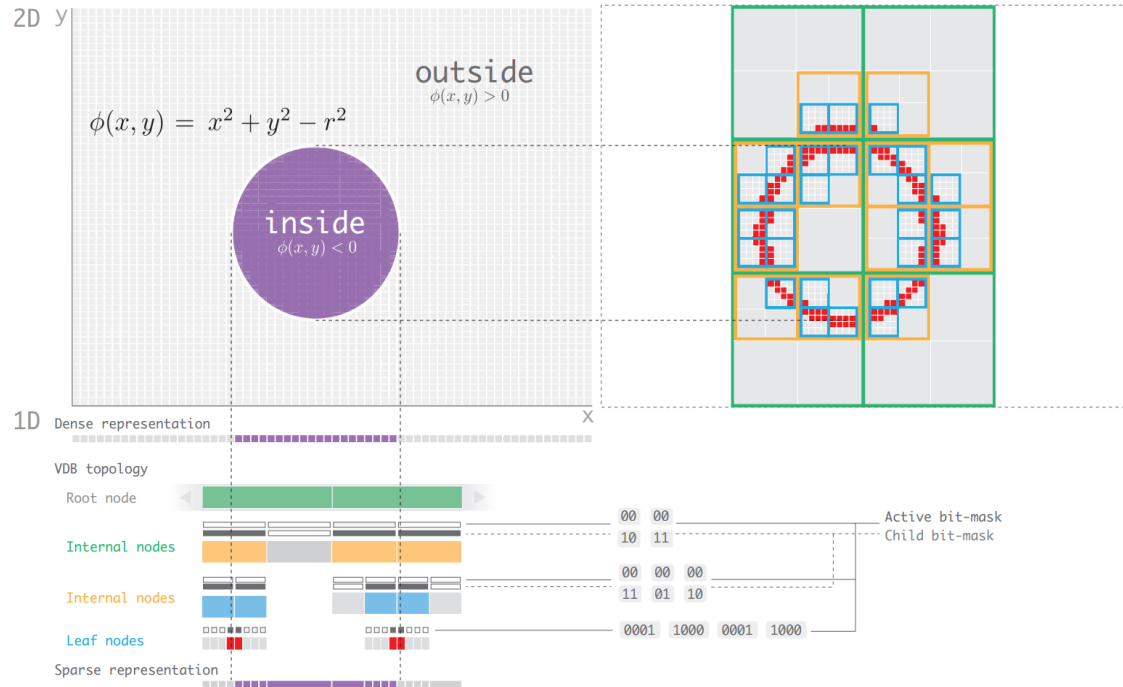
- Root
 - Sparse, resizable
 - Represented as hash map
- Other nodes
 - Dense, restricted to powers of two in size
 - Direct access table
- Any node
 - May contain pointers to values instead of pointer to node
 - Used when all tiles in an internal node have the same (or no) value.

Bit Masks

- Direct access bit masks
 - Embedded in the tree structure
 - Used for
 - Encoding topology
 - Sequential iterators
 - Lossless compression
 - Boolean operations
 - Morphology operations

2D Structure

Only stores active tiles.
Values stored at the
highest possible level.



Data access

- Constant time access
 - Random
 - Constant height
 - Inverted tree traversal (“caching”)
 - Sequential
 - Iterators
 - Stencil
 - Combination of the above

Application algorithms

- Topological Morphology Operations
 - Dilation and erosion using the bit masks
- Level set applications
 - Constructive Solid Geometry
 - Boolean Topology Operations
 - Mesh to Level Set conversion
 - Flood-filling

Results

- Mainly confirming that the data structure performs better than alternatives in most cases, while offering the desired features.
- Trades a slightly larger memory usage for additional flexibility compared to DT-grid.
- Has been successfully applied by DreamWorks Animation in film production.