CSE-170 Computer Graphics

Lecture 23

Spatial Decomposition Schemes and Constructive Solid Geometry

Dr. Renato Farias
rfarias2@ucmerced.edu

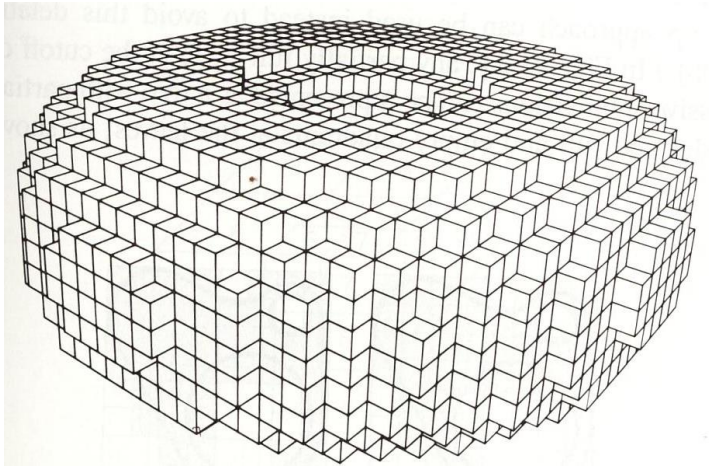
Spatial Decomposition Schemes

Decomposition Models

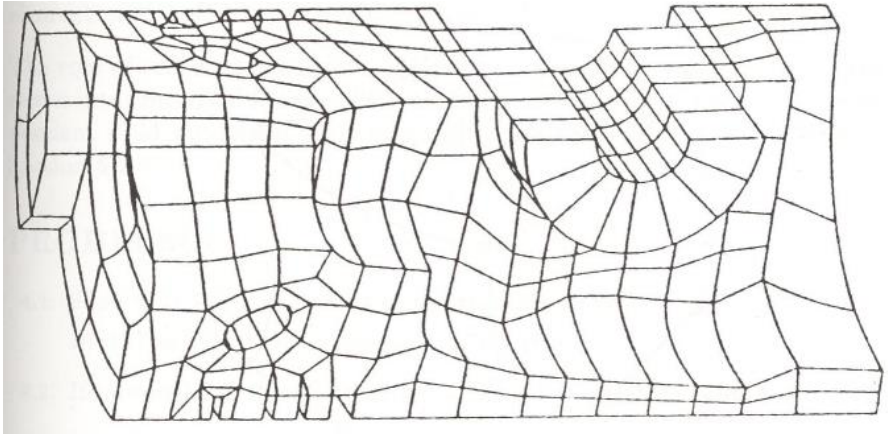
- Exhaustive Enumeration
  - Irregular cell decomposition
  - Regular subdivision
    - Ex: tetrahedra, grids, etc

Decomposition Models

- Regular Decomposition



- Irregular Decomposition also possible
  - Can lead to perfect boundaries



Decomposition Models

- Exhaustive enumeration is not compact…

- Adaptive subdivision!
  - Quadtrees
  - Octrees
  - BSP trees

=>be sure you know these three structures !

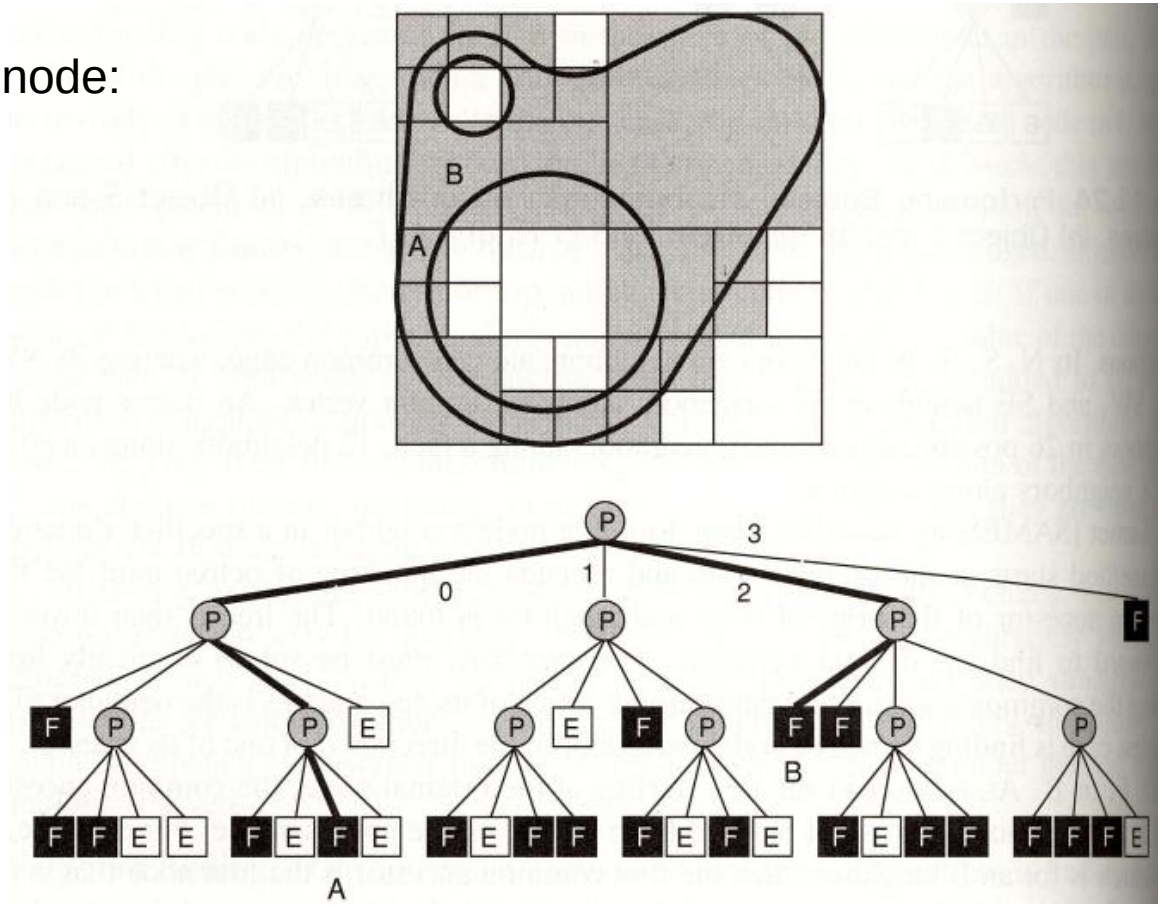
Quadtrees and Octrees

Quadtrees

- Example:

Children order for each tree node:

2	3
0	1



Color of tree nodes:

- Black: cell occupied
- White: empty cell
- Gray: hybrid occupation (thus needs subdivision)

Quadtrees

- Quadtrees are compact
  - In general, number of nodes is proportional to the object's perimeter

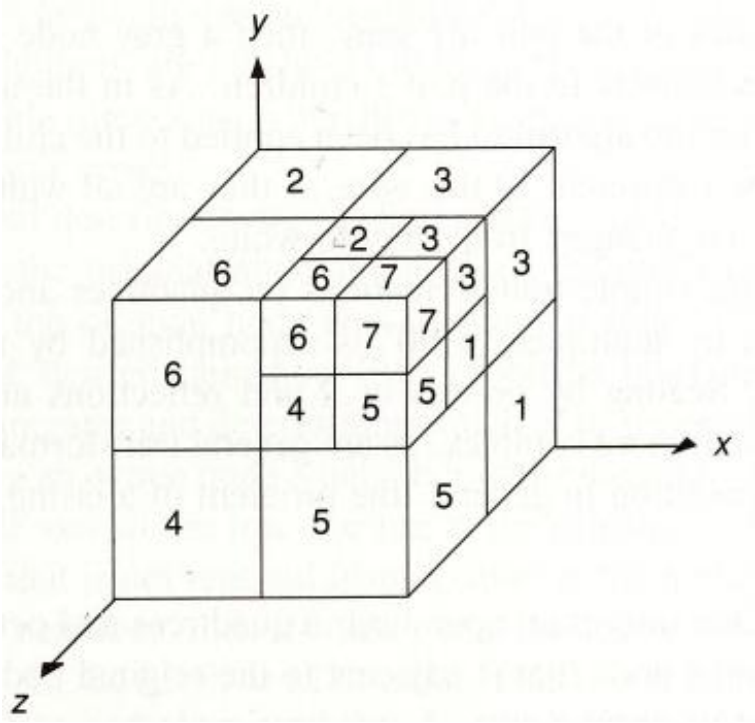
- Advantages
  - Very popular to represent large datasets
  - Boolean operations are still efficient

Quadtrees

- Drawbacks
  - Translations and general transformations are difficult to apply
- Neighbor Finding
  - Not trivial
    - Algorithm: ascend its branch until common ancestor, then descend until finding neighbor

Octrees

- Properties are very similar to quadtrees
  - But one extra dimension is added



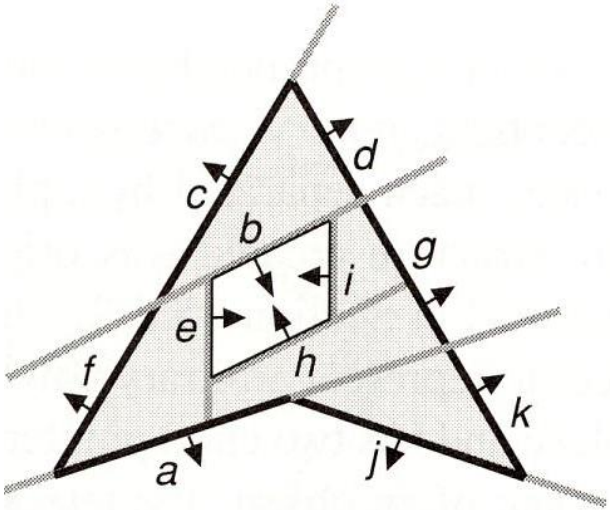
Octrees and Quadtrees

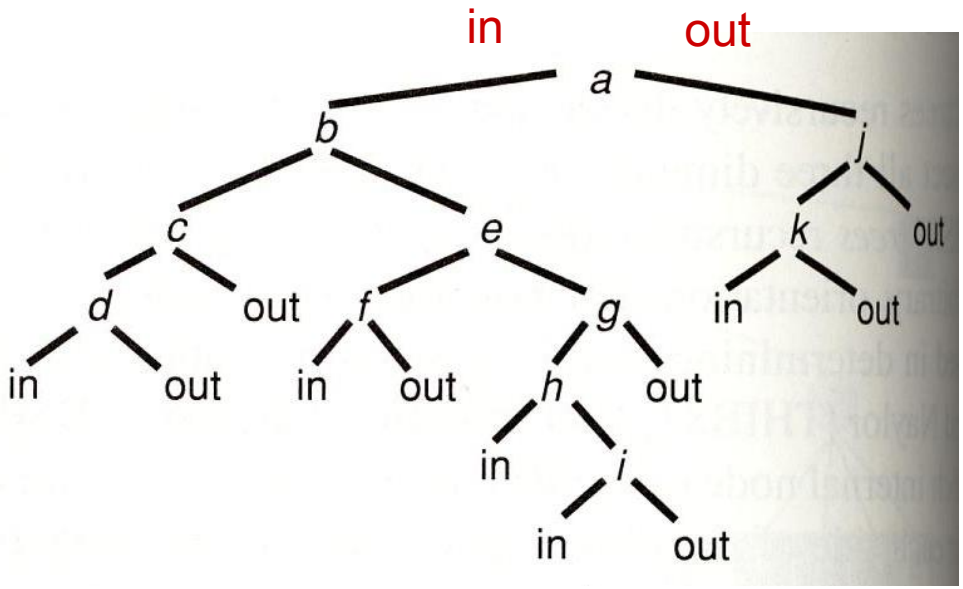
- Several extensions possible:
  - Pointerless notation possible (indices in array)
  - Quadtrees can be simplified to bintrees (one separation axis at a time)
  - Linear notation
    - Node types represented as digits and concatenated
    - Several algorithms available for dealing with the linear notation

BSP-trees

Binary Space-Partitioning Trees

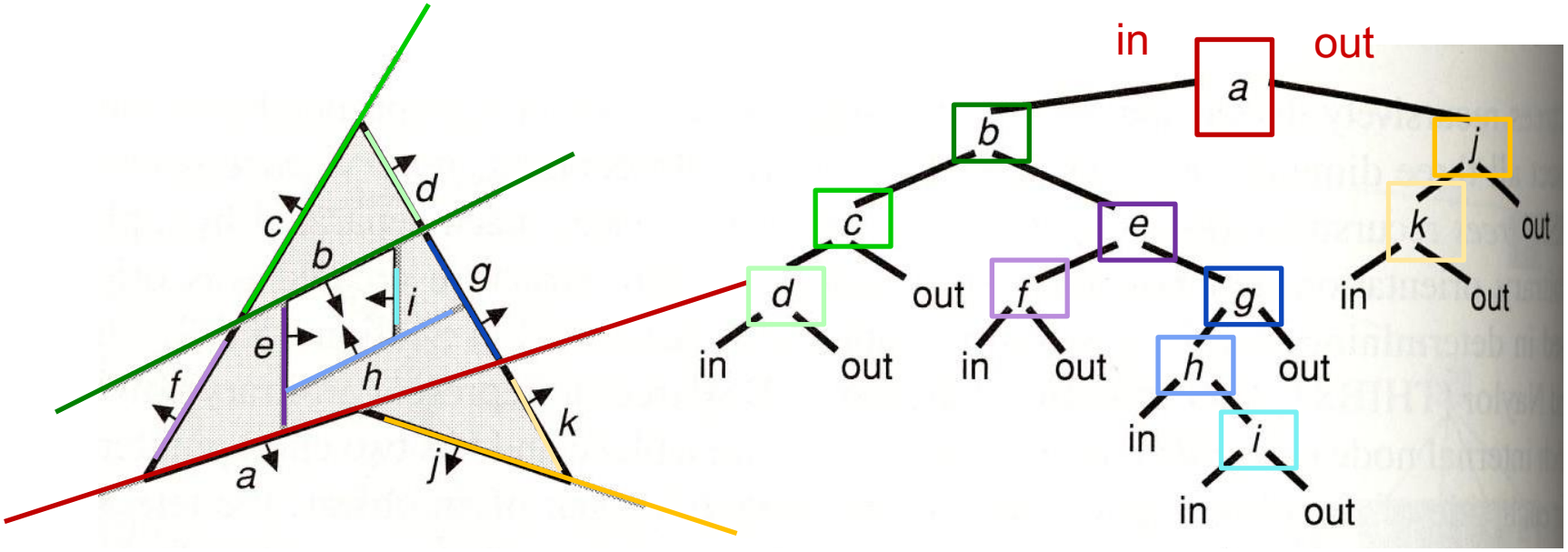
- BSPs can be used for representing the boundary of objects!
  - Example for a polygon:





Binary Space-Partitioning Trees

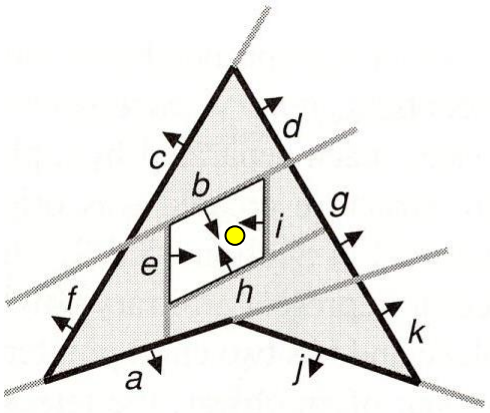
- BSPs :
  - Construction process:

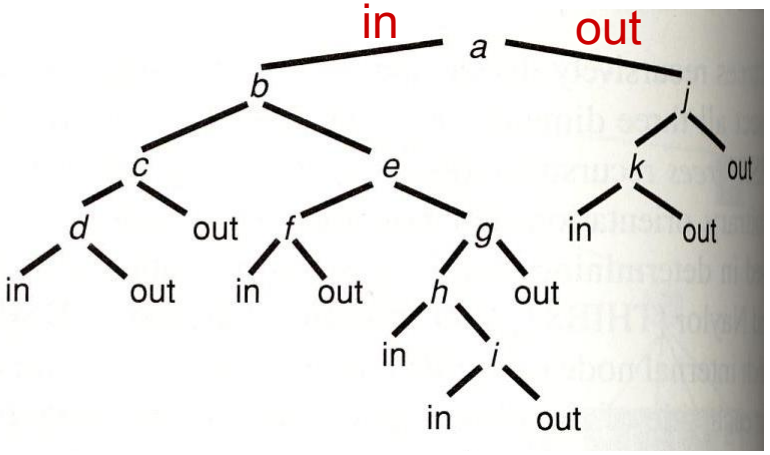


Binary Space-Partitioning Trees

- Point Classification Problem

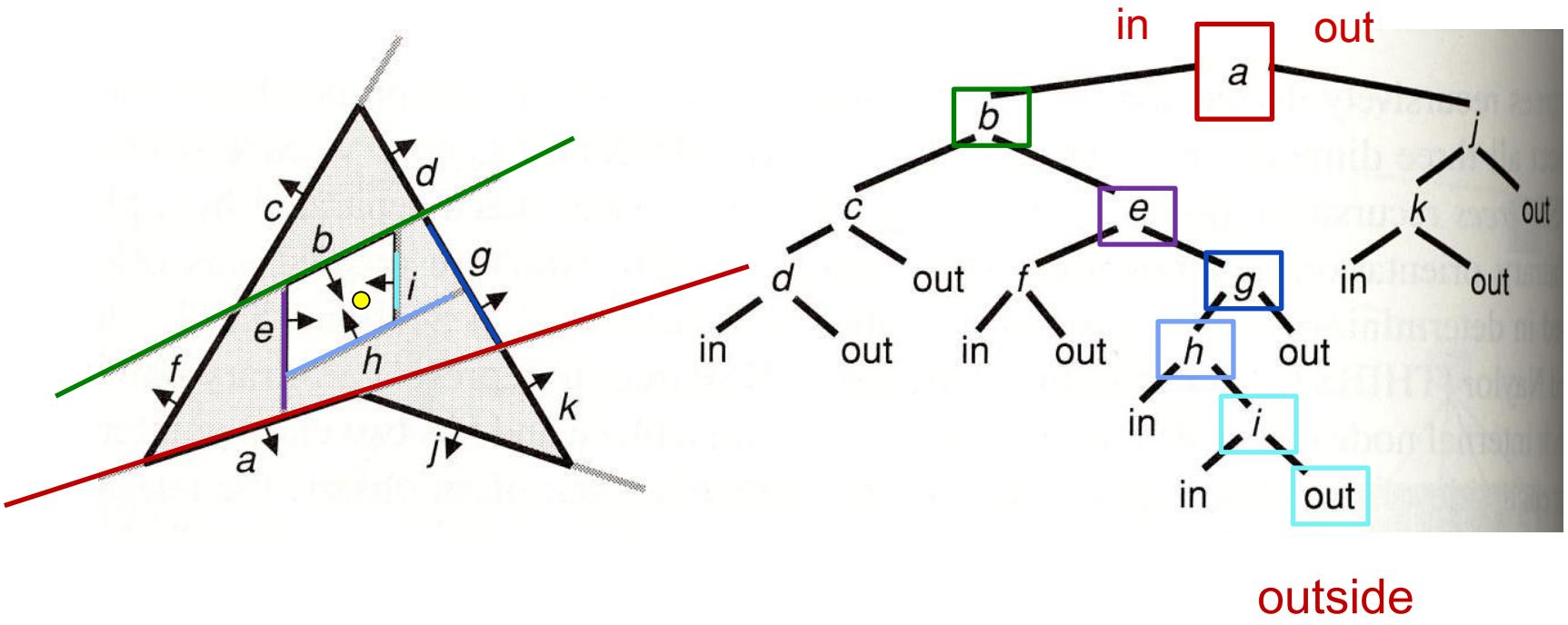
```
classify ( point p, node n )

  if (n is leaf) return leaf state;

  if p out in respect to n
    return classify ( p, n->right );
  else
    return classify ( p, n->left );
```





Binary Space-Partitioning Trees

- Point Classification Problem
  - Example for yellow point:



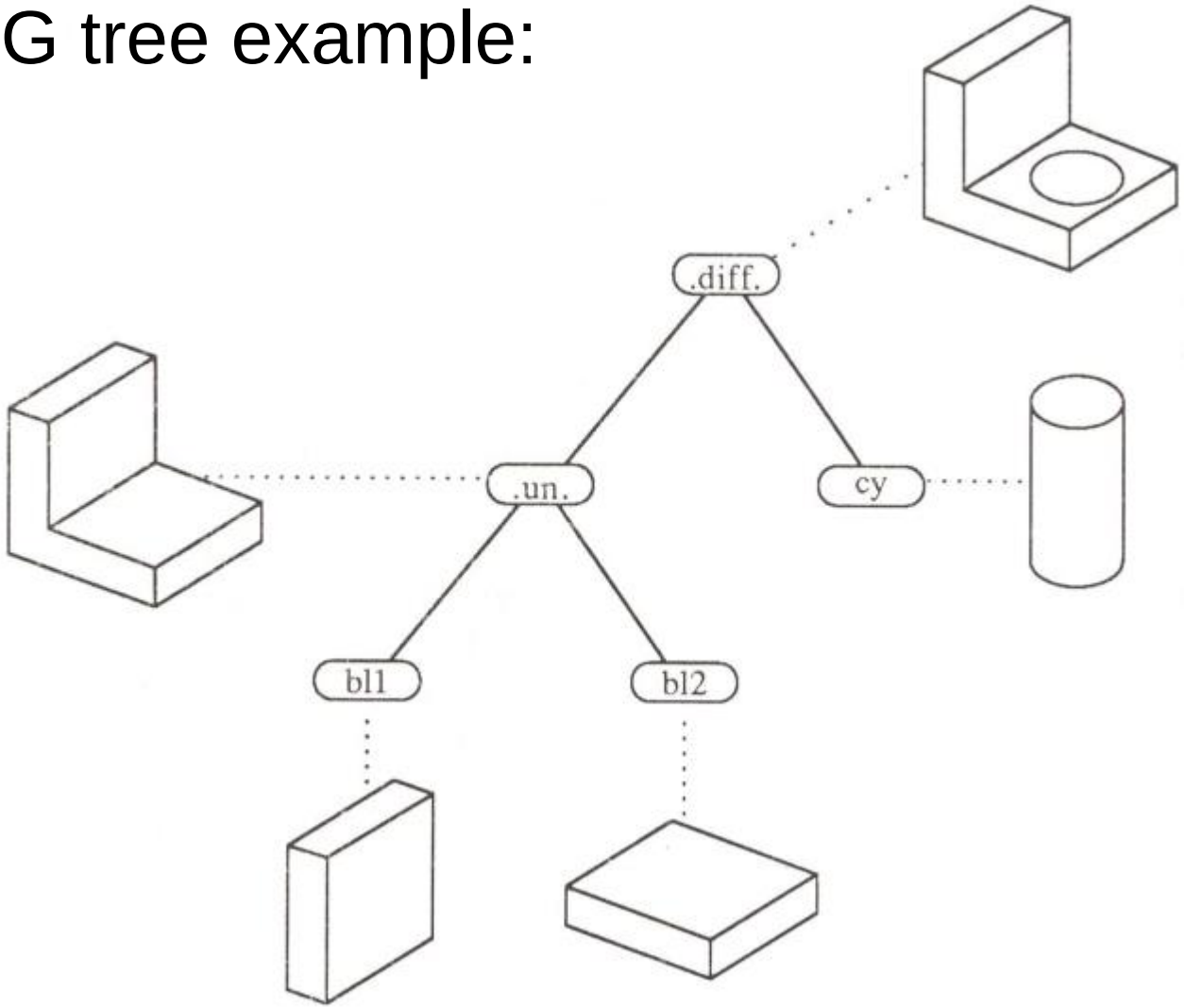
Constructive Solid Geometry (CSG)

Constructive Models

- CSGs
  - Constructive Solid Geometry
  - Popular for mechanical design
  - It is a binary tree:
    - Leafs are primitives
    - Every non-leaf node has an operation to be applied to the two children

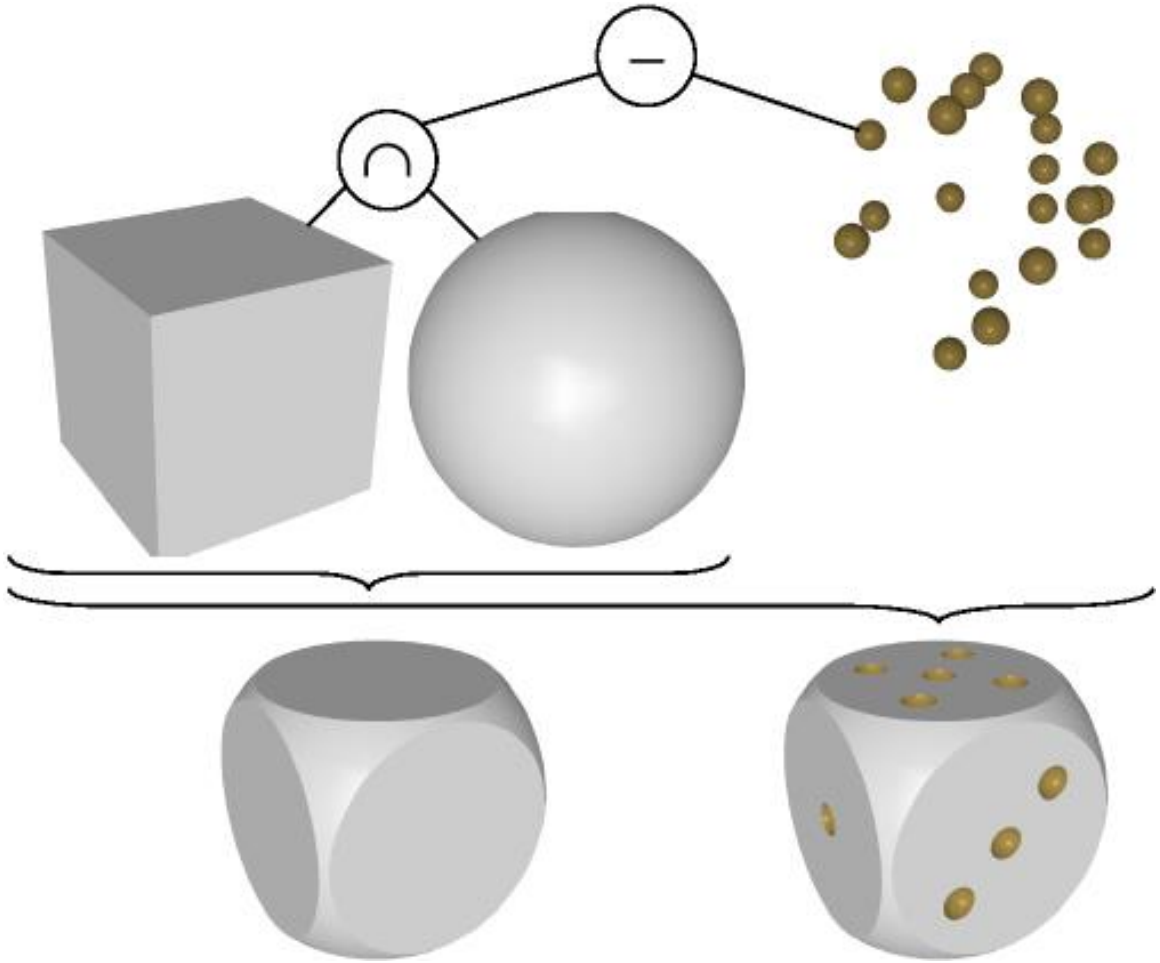
CSG

- CSG tree example:



CSG

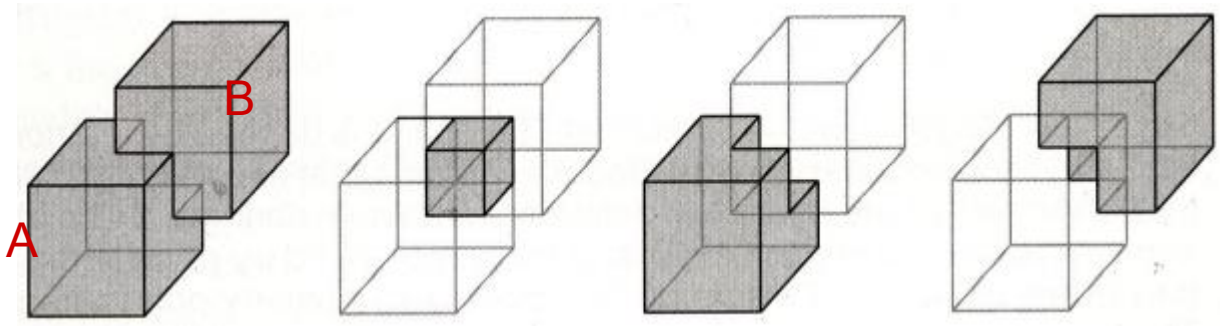
- Example:



CSG: Boolean Operations

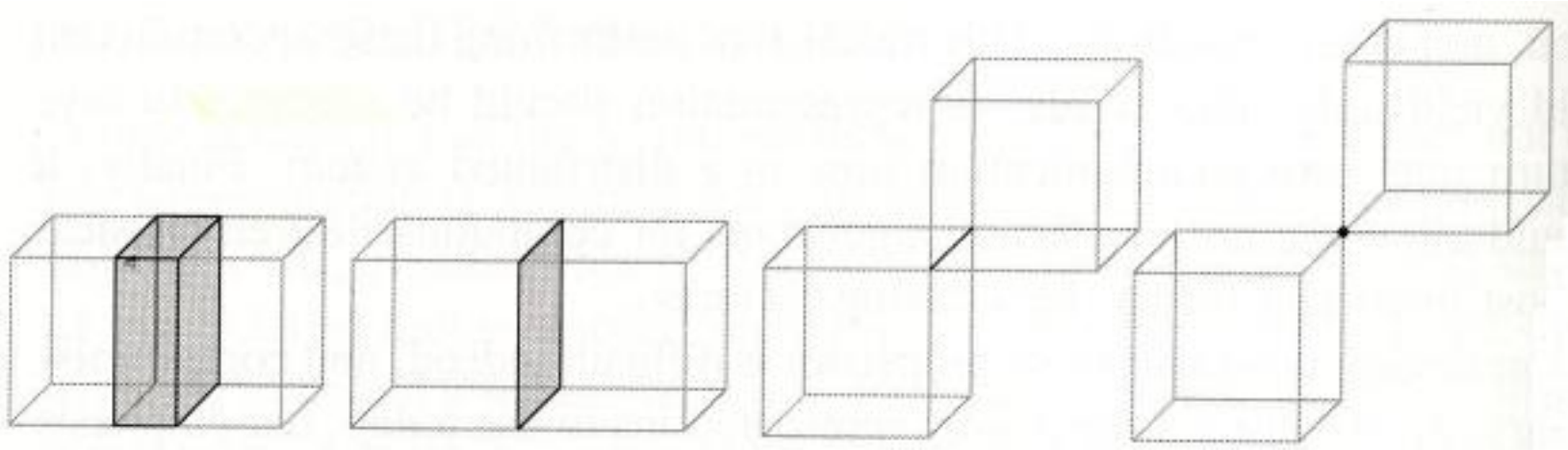
- Ex. of Boolean operation of solid primitives:

AUB A∩B A-B B-A



CSG: Boolean Operations

- Ordinary boolean operations of two cubes may produce:
  - A solid, a plane, a line, a point, or the null set if the two cubes are disjoint



CSG trees

- CSGs are powerful
  - With a good set of primitives: very expressive
  - Compact representation
  - Usually, Boolean operations do not need to be actually computed: most algorithms will run as tree traversals, ex:
    - Point classification
    - Rendering with ray tracers
- Difficulties
  - Determining null objects!
  - Converting to boundary representation