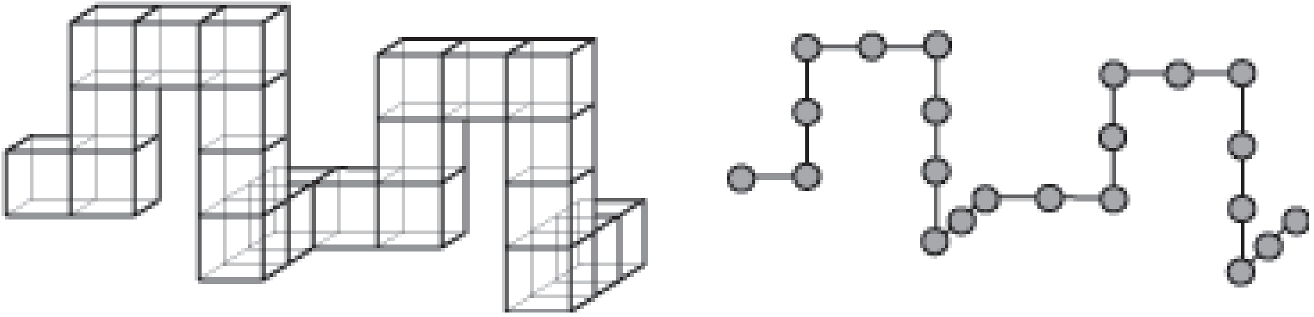
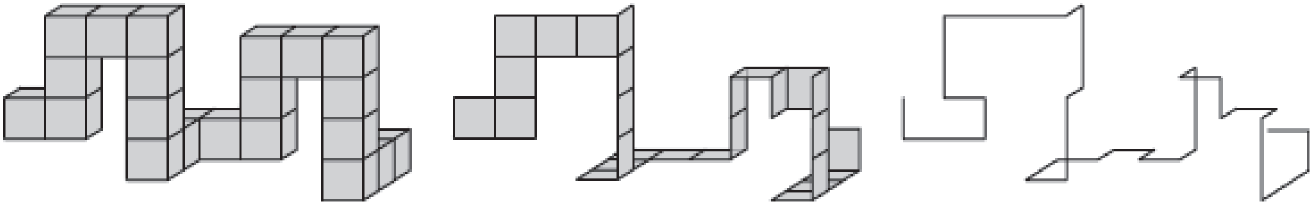
Paths in 3D Pictures



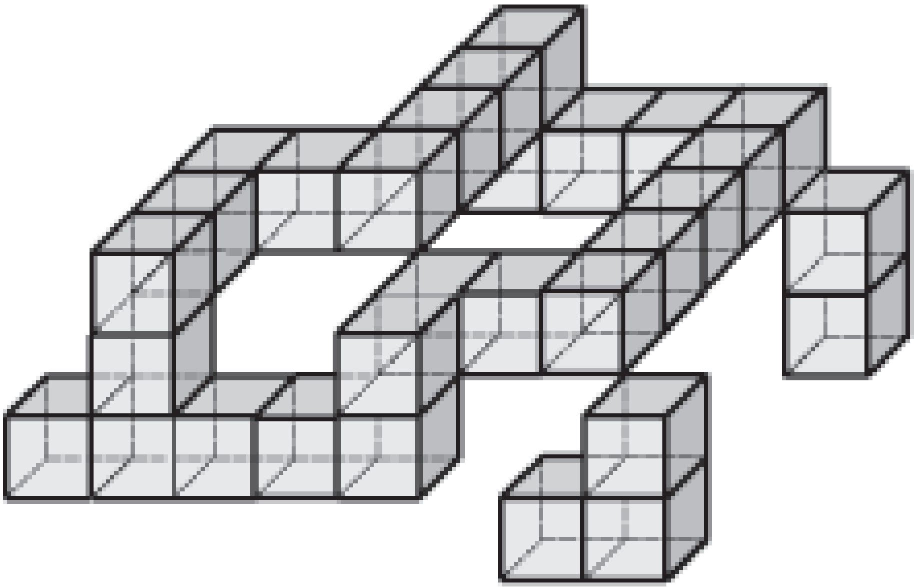
A 2-path in the grid cell model (left) that corresponds with a 6-path in the grid point model (right).



A 2-path of 3-cells (left), a 1-path of 2-cells (middle), and a 0-path of 1-cells (right) in the 3D incidence grid (see page 1 in Lecture 13).

3D: Regions and Components

We recall important basic concepts from Lectures 01, 10, 13, and 14 on 3D pictures. A 3D picture is defined in an $l \times m \times n$ grid $\mathbb{G}_{l,m,n}$, which is embedded into the infinite 3D grid $\mathbb{Z}^3$. Assume the grid cell model:



This is a set of voxels; it consists of one 0-component, two 1-components, and three 2-components.

An *α-region* is a finite α-component. The set in the figure above is a 0-region (or a 26-region, if considered in the grid point model). Note: an assumed infinite α-background of a picture is not a region, but a component.

The FILL algorithm of Lecture 03 can also be applied for labeling components in 3D pictures.

3D: Borders and Boundaries

Let M be a set of voxels.

$p \in M$ is an α-*inner voxel* of M iff $A_\alpha \subseteq M$.

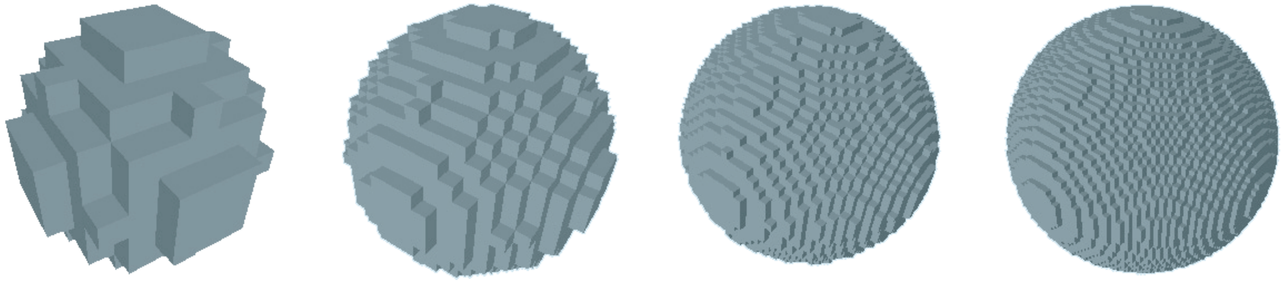
$p \in M$ is an α-*border voxel* of M iff p is not an α-inner voxel of M.

> The set on the previous page only contains border voxels (for any α).

$p \in \overline{M}$ is an α-*coborder voxel* iff $A_\alpha(p) \cap M \neq \emptyset$.

Assuming the grid point model, the α-*boundary* is the set of all invalid α-edges.

Example: Assume a $10 \times 10 \times 10$ cube of voxels. Independent of α, it has $8 \times 8 \times 8$ inner voxels.

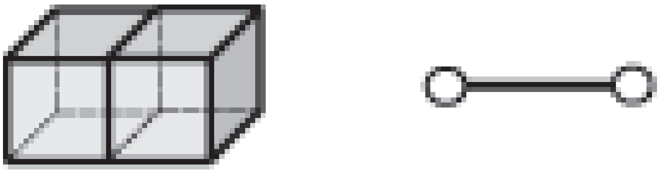


Example: Now assume a digital sphere (say, Gauss digitization in a grid of resolution $1/h$). For reasonably large h, the family of all 2-border voxels is a proper subset of the family of all 1-border voxels, and this is again a proper subset of the family of all 0-border voxels. The set of all 0-border voxels forms a 2-region.

Duality of Grid Point and Cell Model

(i) A 3-cell in the cell model corresponds to its center point, which is a grid point in the grid point model. A 3-cell or a grid point are two options for representing a voxel.

(ii) A 2-cell in the cell model, incident with two voxels p and q, corresponds to the grid edge in the grid point model which connects p and q.



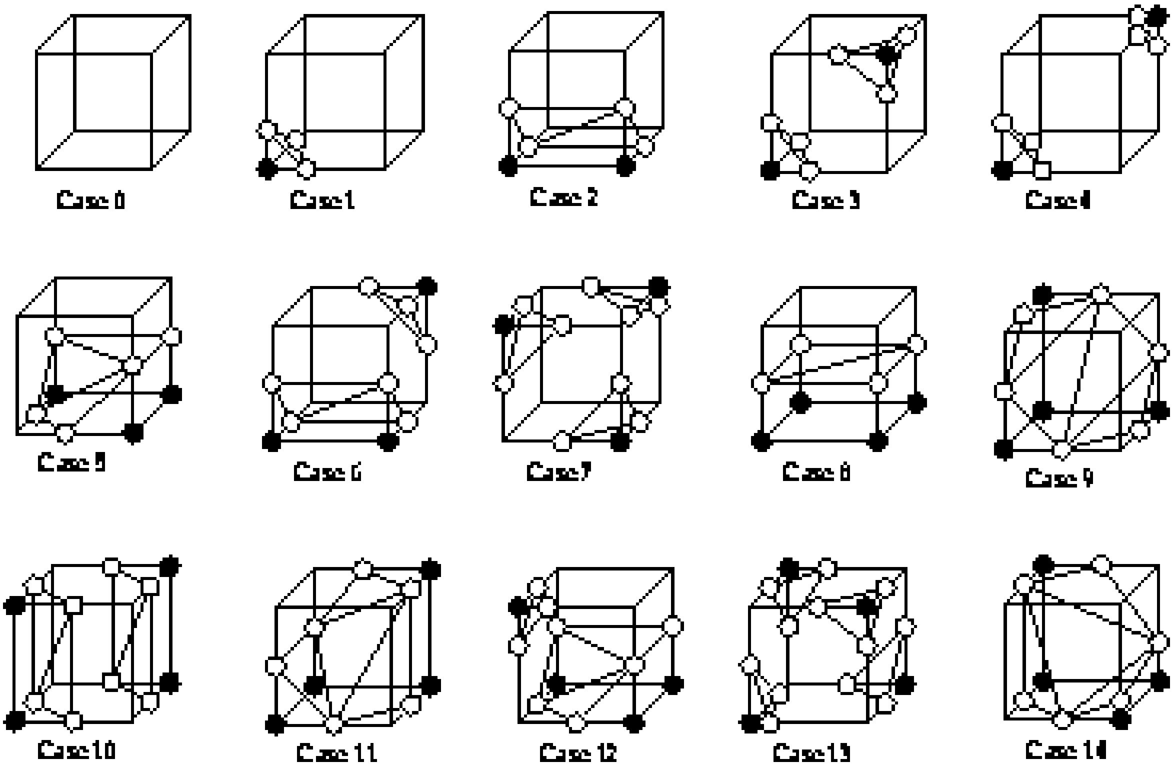
It follows that the α-boundary corresponds to a set of 2-cells. In fact, these are the 2-cells which define the frontier in the grid cell topology.

Frontiers in the cell model have in general the benefit that topological situations are easier to understand (compared to boundaries in the grid point model).

Marching Cubes

For example, the popular “marching cubes algorithms” which construct triangulated surfaces with an intention that these are “cutting” all invalid edges, have the basic problem that they do not always lead to simple closed surfaces [i.e., topologically equivalent to the surface of a sphere] even if the frontier of the given set of voxels is such a simple closed surface.

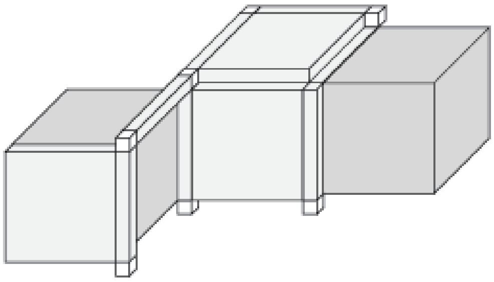
Example of a look-up table:

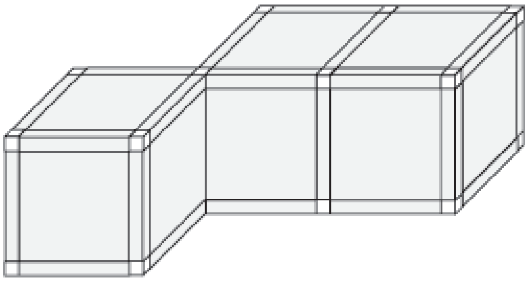


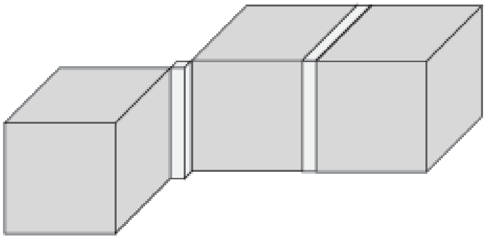
3D Frontiers

Virtual 0-, 1-, or 2-cells are used to define connectedness, frontiers, and open or closed sets.

The frontier of a region M is formed by all those m-cells ($m = 0, 1, 2$) which are both incident with a voxel (3-cell) in M, and with a voxel in $\overline{M}$ (i.e., which is not in M).







Remember the grid cell topology (Lecture 13). The upper set is neither closed nor open. In the bottom row, we have a closed (on the left) and an open (on the right) set. The difference set between closed and open set is the frontier, and it contains 2-cells, 1-cells, and 0-cells.

Equivalence Theorems in 3D and nD

Theorem 1 *For any 3D binary picture, the use of grid cell topology can be replaced by the dual use of 6- and 26-adjacency such that the resulting families of components of white or black voxels are identical in both cases.*

The first equivalence theorem in 2D, and this equivalence theorem in 3D can easily be generalized to nD, $n \geq 2$, where we have the dual use of $2n$- and $(3^n - 1)$-adjacency; see tutorials. For the generalization of the concept of s-adjacency, from 2D to 3D, see the appendix of Lecture 10 (which allows to state an equivalence theorem similar to the second equivalence theorem in 2D); s-adjacency can be generalized to nD by considering space-filling truncated hypercubes, and (for representing 0-cells) regular polyhedra with 2^n faces.

Grid cell topology and the **Maximum-Value Rule** (see page 4 in Lecture 13) provide a way to implement “connected” and “separated” as dual concepts for nD multilevel pictures (we assume a total order of all voxel values):

> a virtual m-cell c, with $m < n$, is labeled by the maximum value of all voxels (i.e., n-cells) which are incident with c.

Consider the union of all cells (e.g., for $n = 3$ use the geometric representation as on page 3 in Lecture 13) having the same label. This defines a partition of $\mathbb{R}^n$ into closed subsets which pairwise intersect at frontier segments only.

3D Frontier Grid

0-cells in the 3D picture grid

$$\mathbb{G}_{l,m,n}$$

are mapped into voxels in the 3D frontier grid

$$\mathbb{F}_{l+1,m+1,n+1}$$

(see page 4 in Lecture 14).

Whatever α we assume for a region in the 3D picture grid, its cell representation in the 3D frontier grid will be 2-connected.

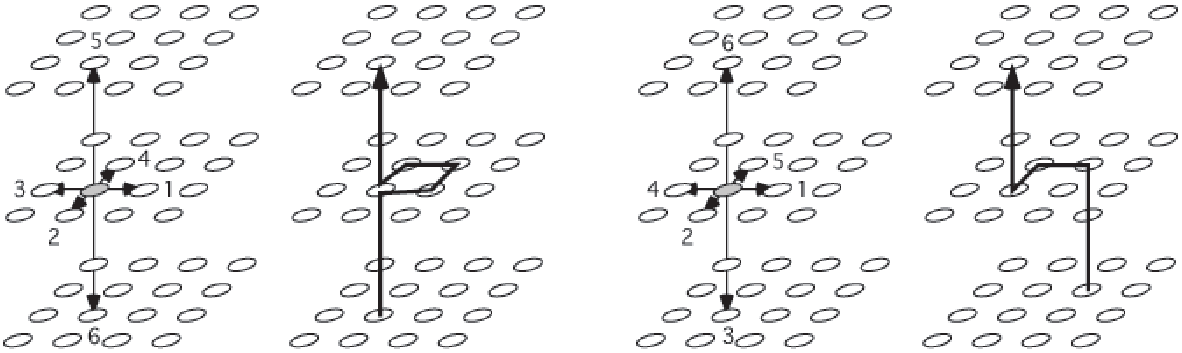
The 3D frontier grid is used as a **“drawing board”** for analyzing individual regions, always just one at a time, and not for representing several regions, or even the whole picture at one moment.

Basically, at first the region under consideration is mapped into $\mathbb{F}$ as being a closed set in the grid cell topology, with one voxel for any 0-cell incident with a region’s voxel in $\mathbb{G}$. However, when analyzing these voxel sets in $\mathbb{F}$ we use the original picture values for understanding local topological situations. (Note: this remains basically the same as for the 2D grid, see Lecture 14).

Local Circular Orders in 3D?

It is possible to introduce local circular orders $\xi(p)$ at voxels p in an adjacency graph $[\mathbb{Z}^3, A_\alpha]$.

For example, consider $\alpha = 6$. We take first all four 6-adjacent voxels in the same layer (defined by a constant z-coordinate) in clockwise order, then the one on top, and finally the one below.

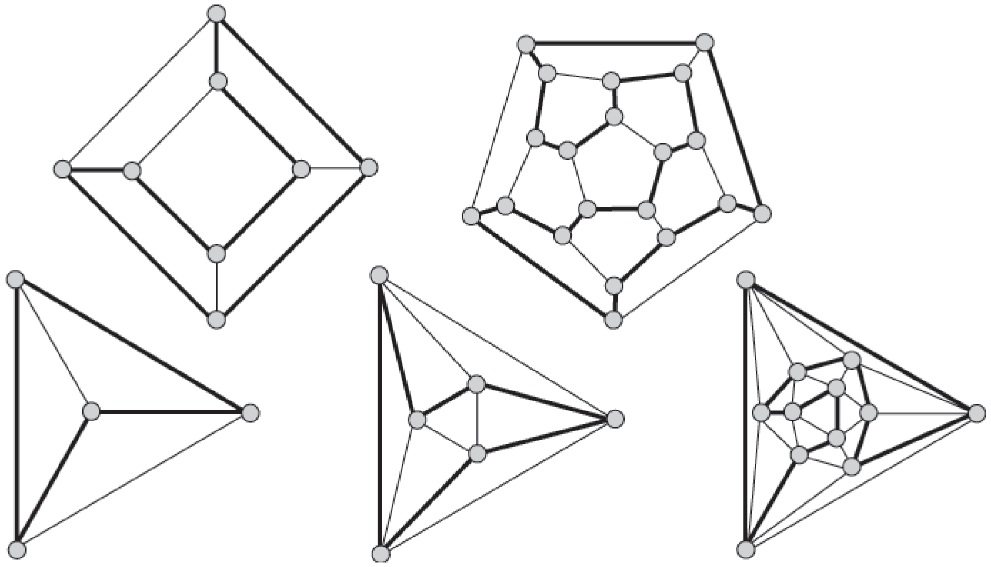


We start with one step from below to the voxel above. According to the clockwise order in the layer of the voxel, we pass now through an elementary 4-cycle in that layer, which we leave to the pixel in the layer above. This results into an infinite path (!).

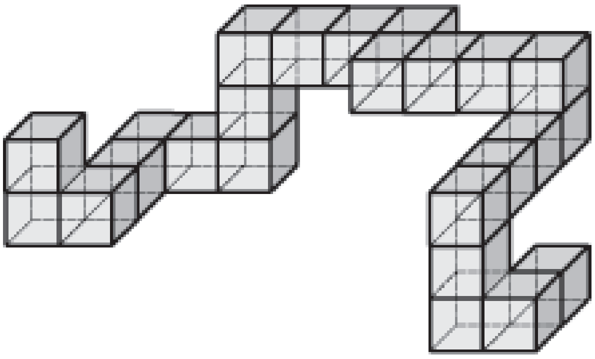
The figure shows on the right one more example.

Hamilton Path

A path that visits each node of a graph exactly once is called *Hamiltonian*. For example, the infinite adjacency graphs $[\mathbb{Z}^2, A_4]$ or $[\mathbb{Z}^3, A_6]$ both have a Hamiltonian path (as considered by G. Cantor in the 19th century).



A graph that has a Hamiltonian circuit is called a *Hamiltonian graph*. Of course, the graph needs to be finite in this case.



The 2-cells of the frontier of a simple 2-arc of 3-cells have a Hamiltonian path with respect to 1-adjacency.

The Need for a 3D Concept

Adjacency graphs $[\mathbb{Z}^3, A_\alpha]$ ($\alpha = 6, 18, 26$) do not allow to introduce a local circular order such that directed edges always initiate a (finite) cycle.

In short: in case of 3D pictures we cannot apply the concept of local circular orders.

A cycle through all 2-cells of a frontier would define a Hamiltonian path in the 1-adjacency graph of all 2-cells of a frontier. However, such a path does not exist in general.

(**Open problem;** Can we generalize the statement above, for example in the following way: In general we have that any adjacency graph $[\mathbb{Z}^3, A]$, which cannot be embedded into an adjacency graph $[\mathbb{Z}^2, B]$ (e.g., as possible if A just defines one infinite path [a "Cantor scan"] through all voxels in $\mathbb{Z}^3$), does not allow to introduce a local circular order such that directed edges always initiate a [finite] cycle.)

Coursework

Related material in textbook: Sections 2.2.1 and 4.2.1.

A.25. [6 marks] Assume that we know that our input is a simple 2-arc of 3-cells, given by one of its two endvoxels p_1 and a sequence of voxels p_2, p_3, …, p_m such that p_{i+1} is 2-adjacent to p_i, for $i = 1, \ldots, m-1$. (The arc is *simple* iff any of the voxels p_2,...,p_{m-1} is only 2-adjacent to exactly two other voxels in the arc, and any of the two endvoxels p_i and p_m is only 2-adjacent to exactly one other voxel in the arc.)

Write a program which calculates a Hamiltonian path of all 2-cells (with respect to 1-adjacency of these 2-cells) of the frontier of an arbitrary simple 2-arc of 3-cells. In your accompanying text:

(i) Start with discussing a single 3-cell, which is a simple 2-arc of (graph-theoretical) length 1.

(ii) Explain your representation of 2-cells in $\mathbb{Z}^3$.

(iii) Specify your choice of directional steps at “corner voxels” of a 2-arc.