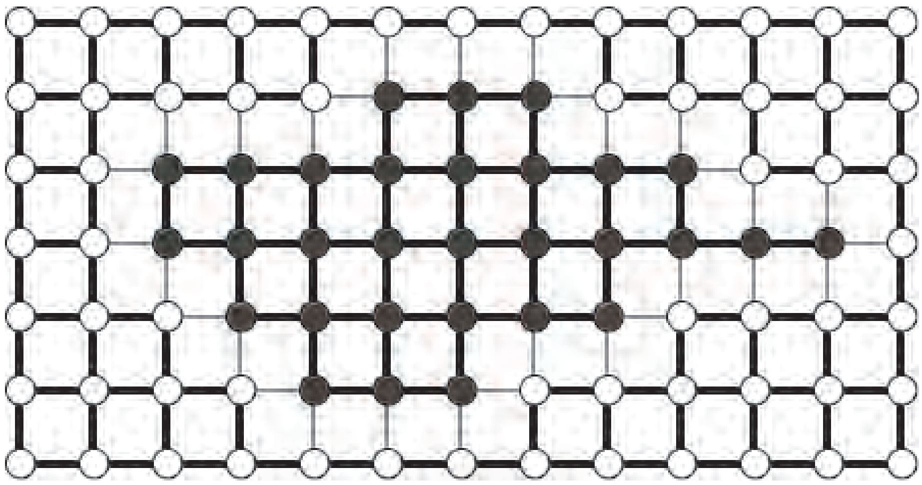
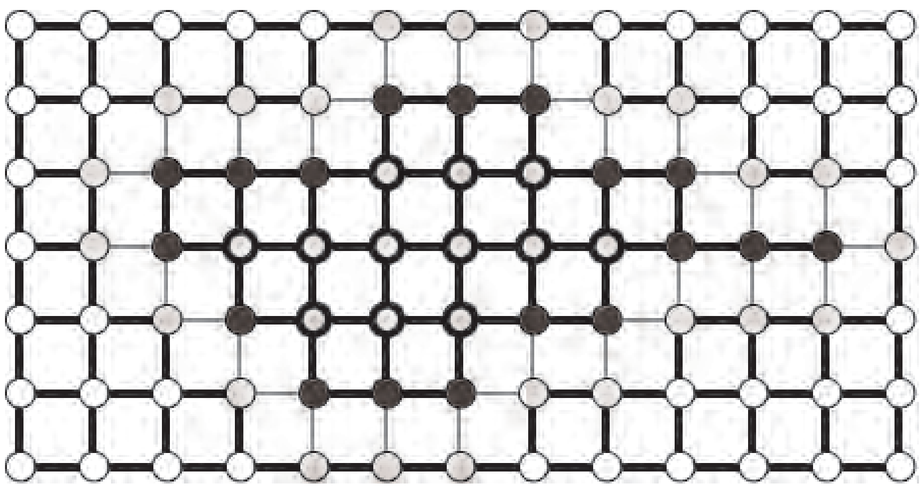
4-Border and 4-Boundary



set S = black and white pixels; set $M \subseteq S$ = black pixels

invalid edges = all edges between M and $\overline{M} = S \setminus M$



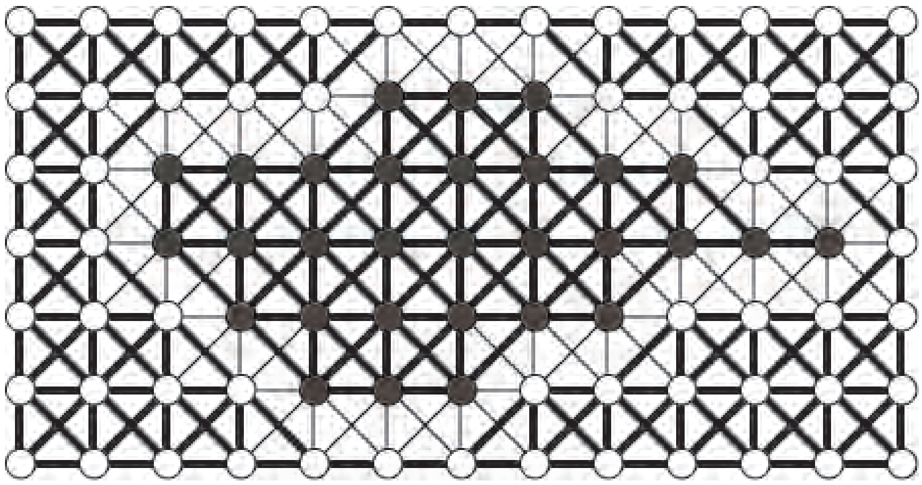
$p \in M$ *4-inner pixel* iff $A_4(p) \subseteq M$ (shown in gray)

$p \in M$ *4-border pixel* iff p not a 4-inner pixel (shown in black)

$p \in \overline{M}$ *4-coborder pixel* iff $A_4(p) \cap M \neq \emptyset$ (shown in gray)

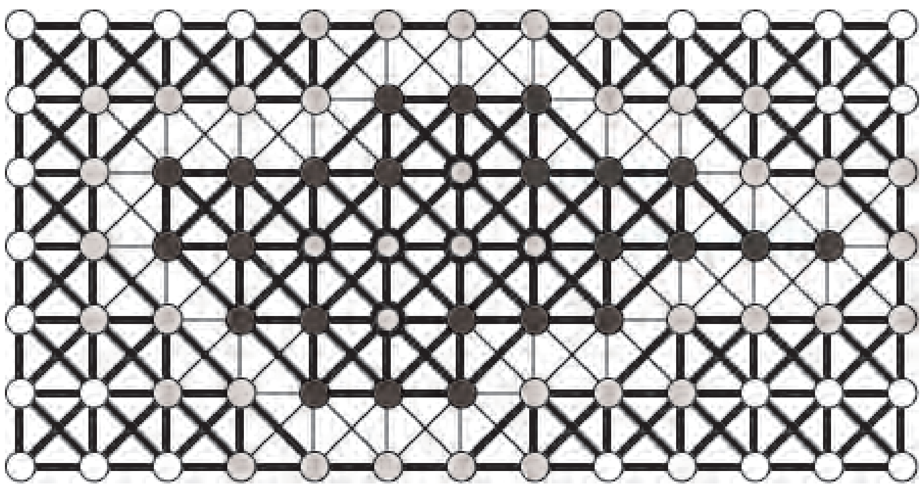
4-boundary of M = set of all invalid edges

8-Border and 8-Boundary



invalid edges = all edges between M and $\overline{M} = S \setminus M$

(note: 8-adjacency also defines diagonal edges)



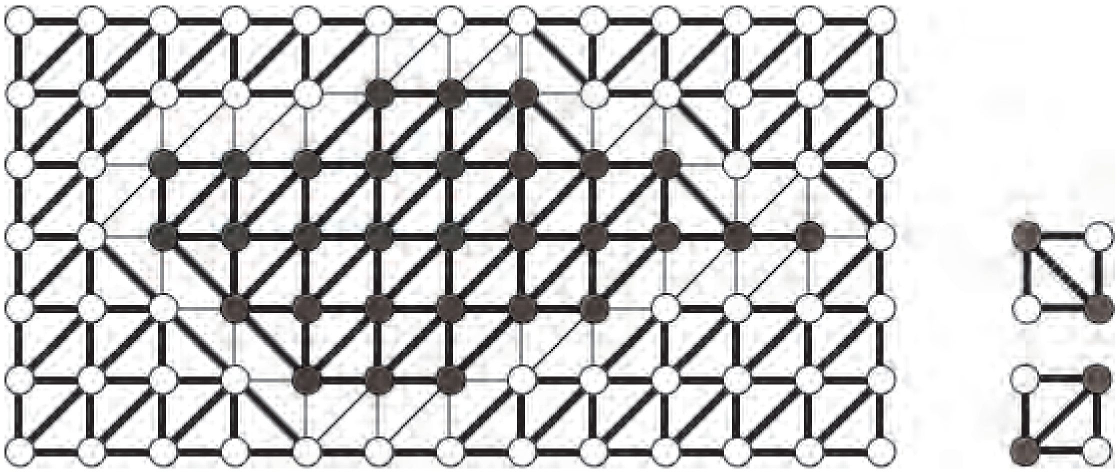
$p \in M$ *8-inner pixel* iff $A_8(p) \subseteq M$ (shown in gray)

$p \in M$ *8-border pixel* iff p not a 8-inner pixel (shown in black)

$p \in \overline{M}$ *8-coborder pixel* iff $A_8(p) \cap M \neq \emptyset$ (shown in gray)

8-boundary of M = set of all invalid edges

s-Border and s-Boundary

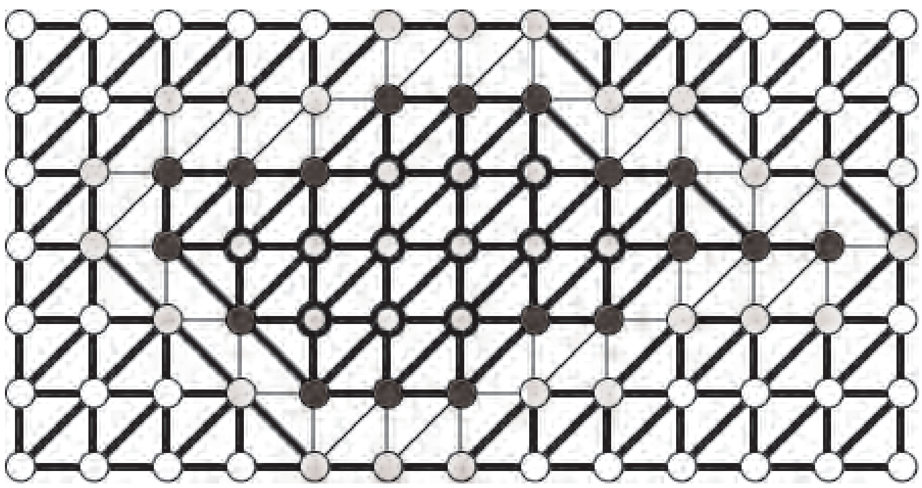


s(witch)-adjacency: one diagonal edge in each 2×2 square

1. three pixel values equal: diagonal connects equal pixels

2. two diagonal pairs of pixels (*flip-flop case*; see both squares on the right): diagonal between preferred values (assume a total order of all picture values)

3. otherwise: lower left to upper right (just to specify a way)



s-inner pixel, *s-border pixel*, *s-coborder pixel*, and *s-boundary*

Undirected Graph

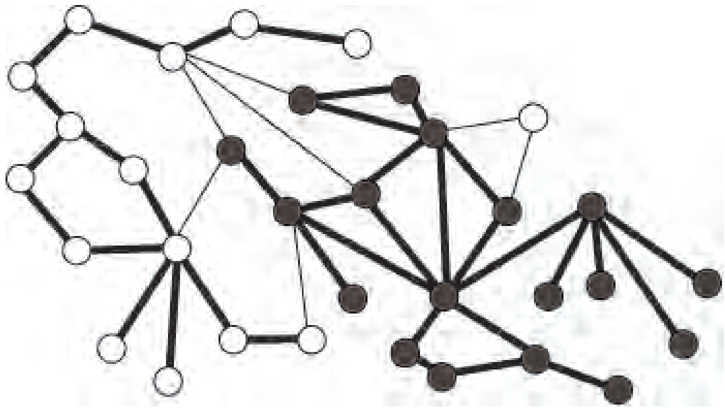
$[S, R]$ is a (simple undirected) *graph* iff S is a set and R is a symmetric and irreflexive relation on S

pRq is equivalent to: "there is an edge $\{p, q\}$ between node p and node q"

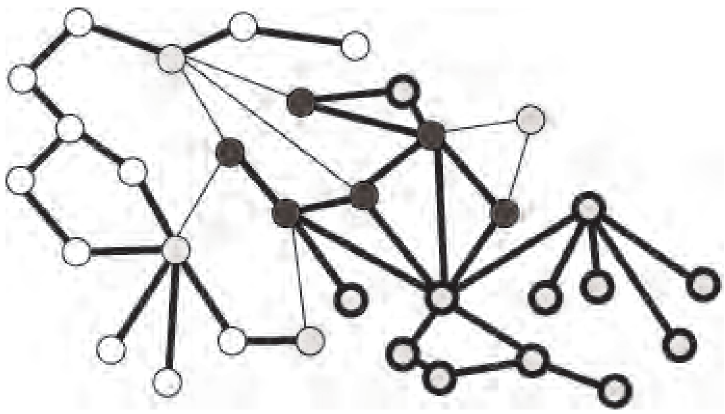
nodes p, q are adjacent iff they are joined by an edge

$[S, R]$ is an *adjacency structure* iff S is countable

examples: $[\mathbb{Z}^2, A_4]$ or $[\mathbb{Z}^3, A_{18}]$



$M \subseteq S$ (black nodes) also defines invalid edges, inner nodes, border nodes, coborder nodes, and a boundary

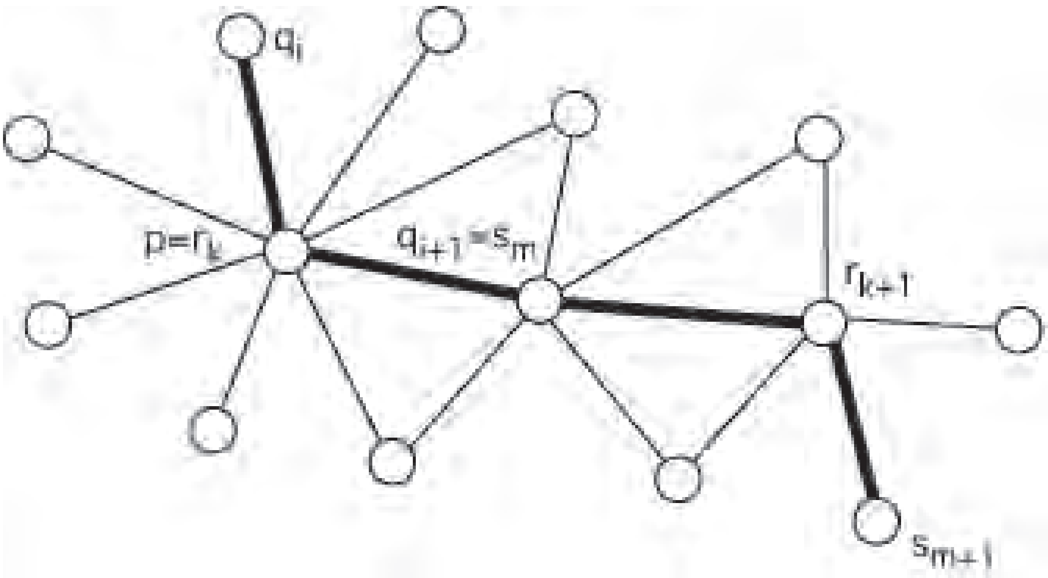


Oriented Adjacency Graph

$[S, A]$ is an *adjacency graph* iff $A(p)$ always finite, S is connected with respect to A, and any finite subset $M \subseteq S$ has at most one infinite complementary component

region = a finite component of an adjacency graph

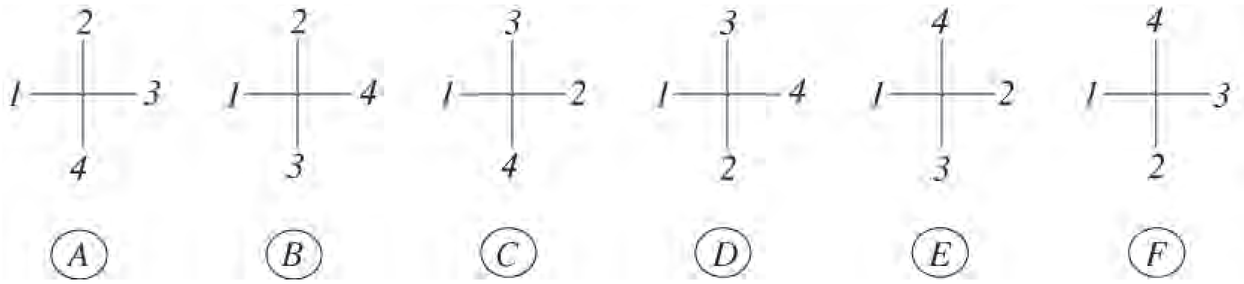
In a *local circular order* $\xi(p)$ at node $p \in S$, the nodes $\langle q_1, \ldots, q_n \rangle$ of $A(p)$ appear exactly once each. We can use these local orders to trace (directed) edges in $[S, A]$ as follows: if we arrive at p from $q_i \in A(p)$, we move next to q_k, where $k = i + 1$ (modulo n).



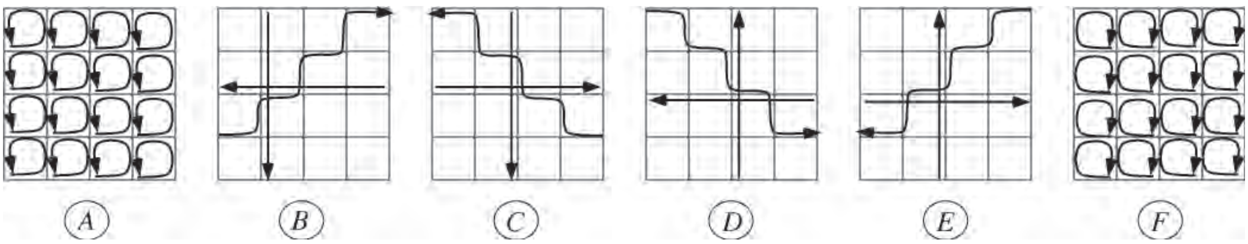
We arrive at q_{i+1}, let $\xi(q_{i+1}) = \langle r_1, \ldots, r_l \rangle$, and $p = r_k$ in this local circular oder of $A(q_{i+1})$. We move next to r_{k+1}. Let $\xi(r_{k+1}) = \langle s_1, \ldots, s_t \rangle$, and $q_{i+1} = s_m$ in this local circular oder of $A(r_{k+1})$. We move next to s_{m+1}; and so forth.

Oriented adjacency graph $[S, A, \xi]$ iff any directed edge initiates a cycle (and not an infinite path).

Two Options for 4-Adjacency



All possible local circular orders for 4-adjacency.



Initiated 4-paths in the infinite grid point plane. Only two cases (A and F) lead to oriented adjacency graphs.

Two Options for s- or 8-Adjacency

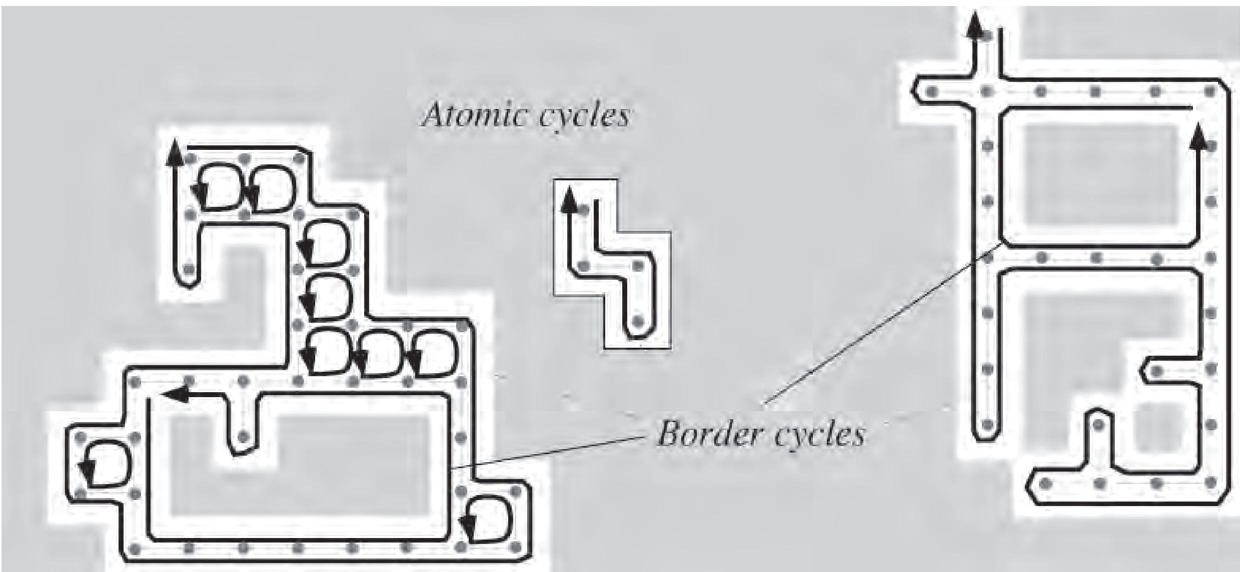
clockwise or counter-clockwise local circular orders define oriented s- or 8-adjacency graphs

Atomic and Border Cycles

A subset $M \subseteq S$ induces a *substructure* $[M, A_M, \xi_M]$ of an oriented adjacency graph $[S, A, \xi]$ where A_M contains only those adjacency pairs $\{p, q\}$ such that $p, q \in M$ and $\{p, q\} \in A$ and where, for any $p \in M$, $\xi_M(p)$ is the *reduced local circular order* defined by deleting from $\xi(p)$ all nodes that are not in M. Such a substructure is an oriented adjacency graph iff M is connected with respect to A_M.

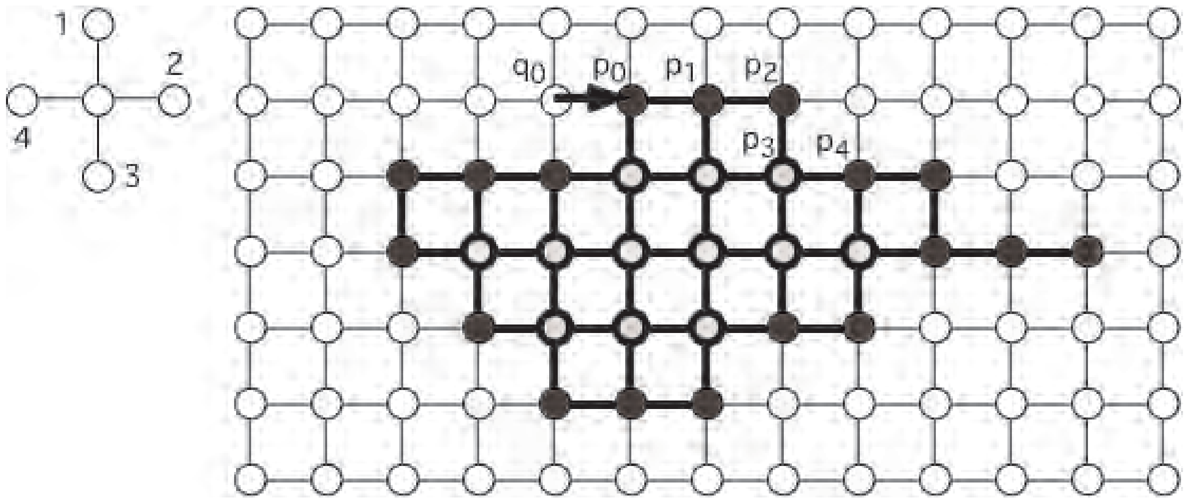
The cycles of $[M, A_M, \xi_M]$ may differ from the cycles of $[S, A, \xi]$. Let (p, q) be a directed edge in $[M, A_M, \xi_M]$, let ρ_1 be the cycle generated by (p, q) in $[M, A_M, \xi_M]$, and let ρ_2 be the cycle generated by (p, q) in $[S, A, \xi]$.

Definition 1 ρ_1 *is an* atomic cycle *iff* $\rho_1 = \rho_2$ *and a* border cycle *otherwise.*

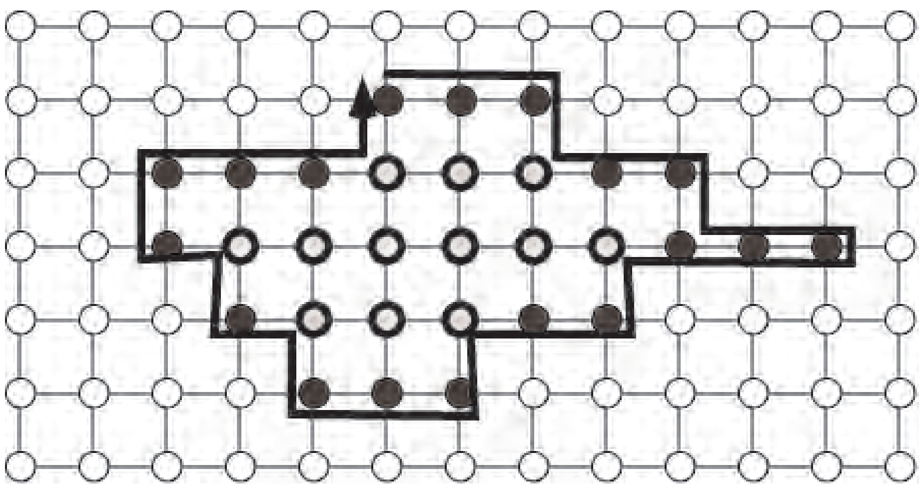


Tracing a 4-Border Cycle

Given: *directed invalid edge* (q, p) from $q \in \overline{M}$ to $p \in M$; let $(q_0, p_0) := (q, p)$ and assume the local circular order as shown.



Let $\xi(p_0) = \langle \ldots, q_0, q^\star, \ldots \rangle$ be the local circular order at p_0. Point $q^\star$ is the pixel above p_0 which is not in M. We take the next pixel in $\xi(p_0)$, which is the pixel right of p_0: this is in M, and it is the next pixel on the border cycle.



Stop: back to original directed invalid edge (p, q) (to step $p_0 p_1$)

General Border Tracing Algorithm

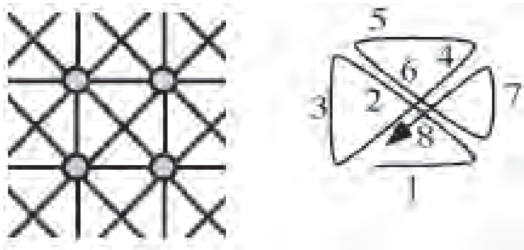
given: oriented adjacency graph; an directed invalid edge (p, q) pointing to a border cycle.

1. Let $(q_0, p_0) := (q, p), i := 0$, and $k := 0$.
2. Let $\xi(p_i) = \langle \ldots, q_k, q^\star, \ldots \rangle$ be the local circular order at p_i. If $q^\star \in \overline{M}$, go to Step 4.
3. Node $q^\star$ is another node on the border cycle. Let $i := i + 1$ and $p_i := q^\star$. Let $\xi(p_i) = \langle \ldots, p_{i-1}, q^\star, \ldots \rangle$ be the local circular order at p_i. If $q^\star \in M$, go to Step 3; otherwise, let $k := i - 1$, and go to Step 4.
4. If $(q^\star, p_i) = (q_0, p_0)$, go to Step 5. Otherwise, let $k := k + 1$ and $q_k := q^\star$, and go to Step 2.
5. We are back at the original directed invalid edge (q, p). The border cycle is $\langle p_0, p_1, \ldots, p_i \rangle$.

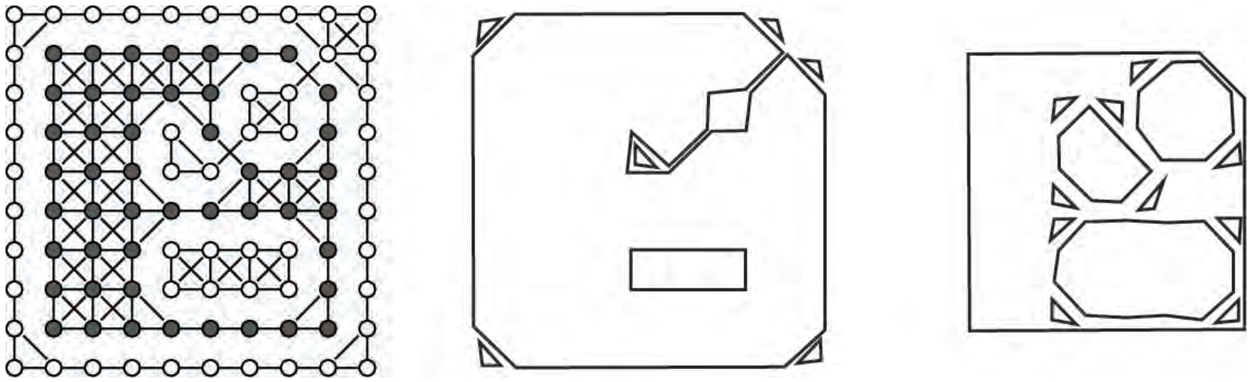
Because (p, q) initiates a cycle (see definition of oriented adjacency graph), the algorithm will always stop.

Border tracing algorithms have been published by many authors (the algorithm above by K. Voss and R. Klette in 1986).

Tracing of an 8-Atomic Cycle



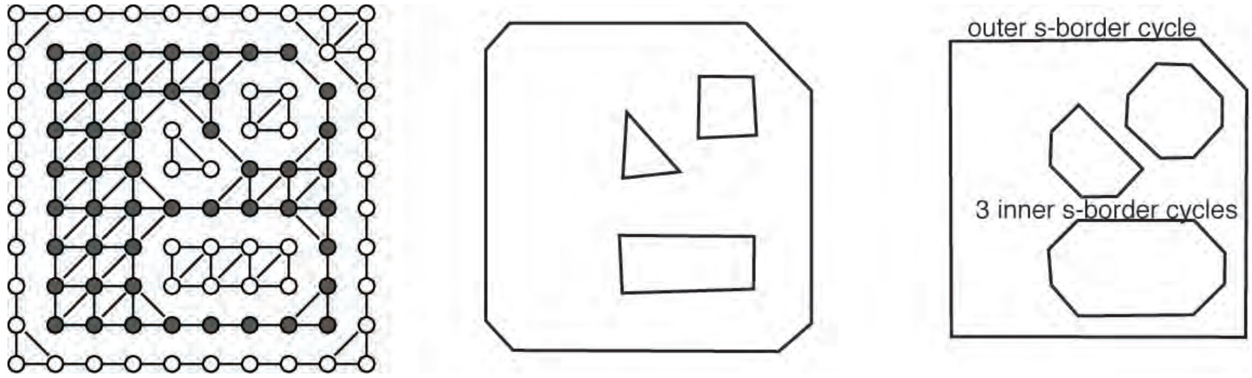
Tracing of 8-Border Cycles



Left: a finite 8-component M in $[\mathbb{Z}^2, A_8]$. Middle: all 8-border cycles of $\overline{M}$. Right: all 8-border cycles of M.

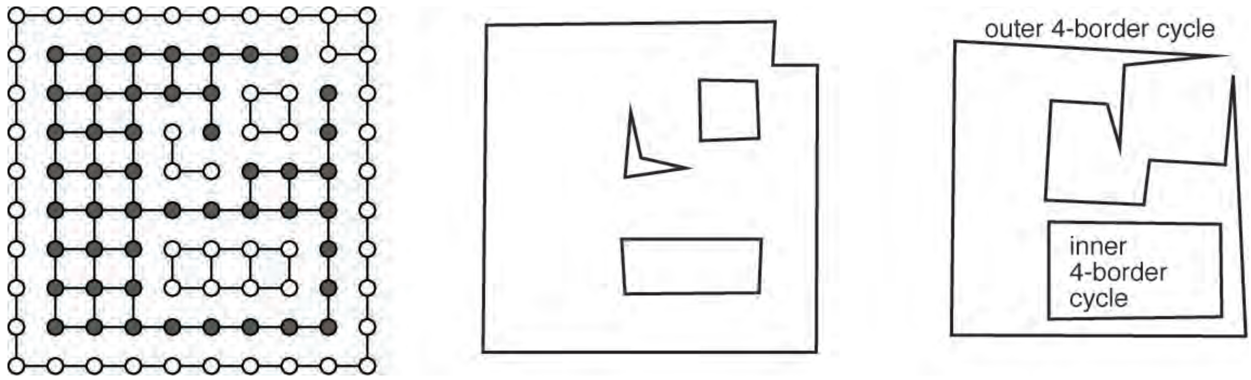
(Compare discussion of reasons for dual use of 4- and 8-adjacency in Lecture 02)

Tracing of s-Border Cycles



Left: a finite s-component M in $[\mathbb{Z}^2, A_s]$ (rules for s-adjacency as on page 3). Middle: all s-border cycles of $\overline{M}$. Right: all s-border cycles of M (note: s-atomic cycles circumscribe triangles).

Tracing of 4-Border Cycles



Left: a finite 4-component M in $[\mathbb{Z}^2, A_4]$. Middle: all 4-border cycles of $\overline{M}$. Right: all 4-border cycles of M (note: 4-atomic cycles circumscribe squares).

Coursework

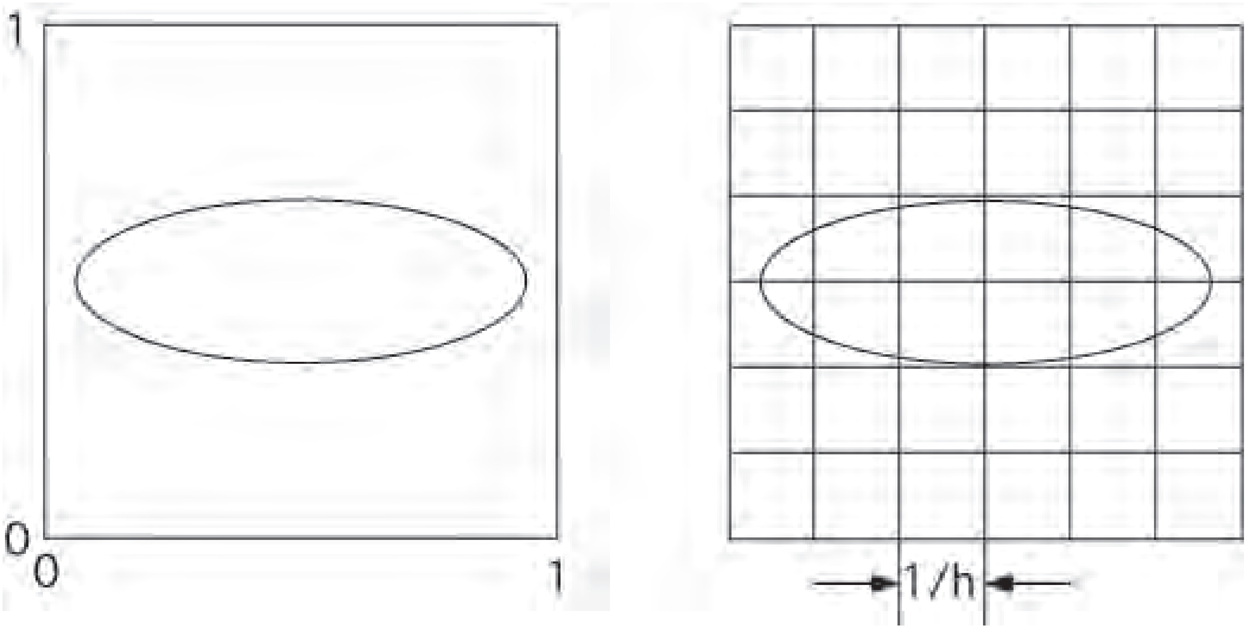
Related material in textbook: Sections 4.1.1, 4.1.3, beginning of 4.1.4, 4.3.1, 4.3.3, and the algorithmic parts of 4.3.4.

A.10. [5 marks] Calculate properties of digitized convex regions (assumed to be 4-regions in binary pictures; see sketch below)

(i) by tracing their (outer) 4-border cycles,

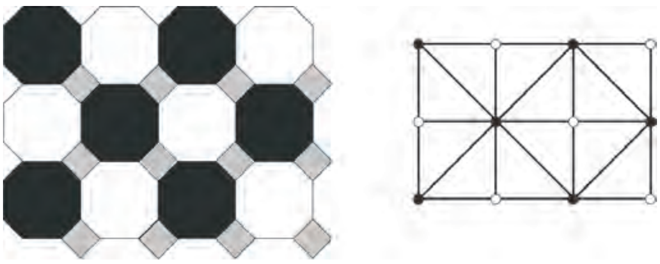
(ii) during tracing update values $\mathcal{P}$ and $\mathcal{A}$ such that these provide estimates for perimeter and area of a given convex region; for $\mathcal{P}$ use the perimeter of the convex hull; you may use Sklansky's algorithm published on page 430 in the textbook (note: this algorithm allows online convex hull calculation during tracing of a 4-border cycle.), and

(iii) discuss how values of *shape factors* $\mathcal{P}^2/4\pi\mathcal{A}$ (see page 28 in textbook) can be used for characterizing your digitized convex regions (you may use different values of grid resolution h).

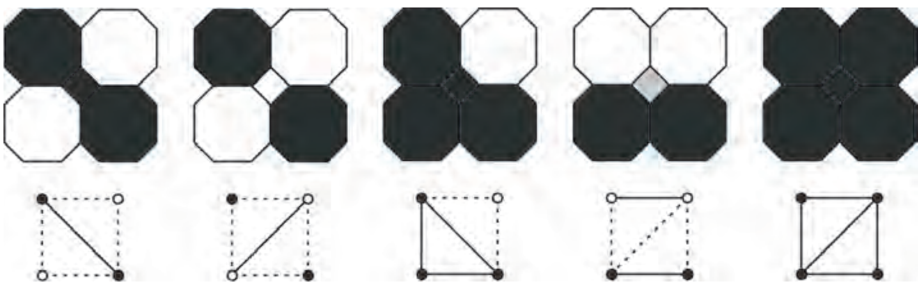


Appendix: Comments on s-Adjacency

(2D) Consider a chessboard-like pattern of white and black squares in the Euclidean plane. The question, whether the white or the black squares are connected, can be answered by considering those points where white and black squares "meet" at corners. For example, if those points are all black, then the black squares are all connected (in the Euclidean topology). – Now consider a chessboard of white or black pixels. The following drawing shows on the left "corner points" as shaded squares, which can be either black or white. On the right, the resulting s-adjacency is shown if all squares are black.

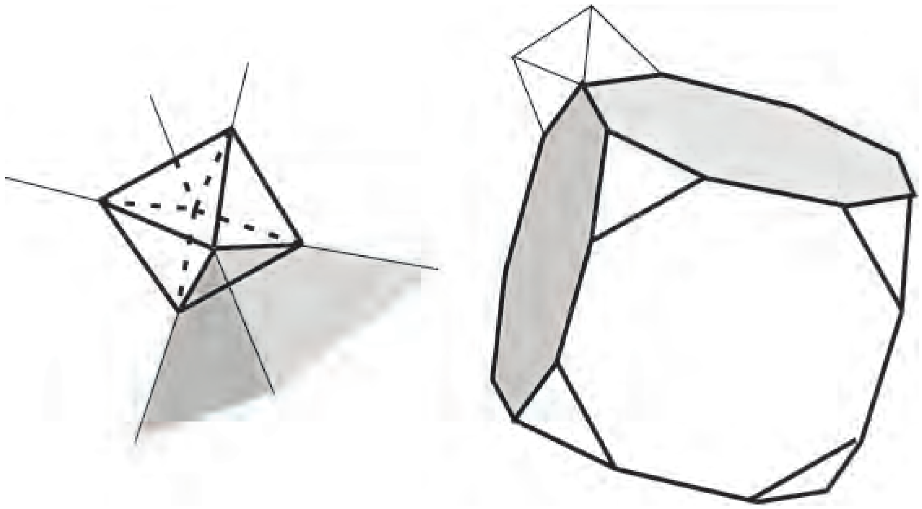


If a 2×2 pattern of white or black pixels represents a flip-flop case, then the diagonal is defined by the color of the (virtual) square representing the corner point. Dashed lines are invalid edges:



Diagonals in the two patterns on the right are by default; we could omit them without impact on connectedness of regions. Note that s-adjacency can be defined for all pictures assuming a total order of their values.

(3D) Consider an alternating pattern of white and black cubes in the Euclidean 3D space: again, corners of cubes decide whether black or white cubes are connected. — For a set of voxels, we represent corner points by regular polyhedrons having eight faces (i.e., an octahedron, composed of equilateral triangles). The drawing shows on the right one shaded voxel (a truncated cube) and on the left an octahedron representing the corner vertex of eight voxels. (These octahedra and truncated cubes are space-filling.)



Each truncated cube is labeled by one 3D picture value. Each octahedron can only have one value (out of the range of all 3D picture values, defined by a total order of all picture values: the ”most important value” is inherited to the face-adjacent octahedron). The resulting face-connectedness between equal-valued (virtual) octahedra and (voxel representing) truncated cubes defines *s-adjacency* in 3D: 6-adjacency between voxels is this way complemented by additional adjacencies.