Algorithms for Drawing Media

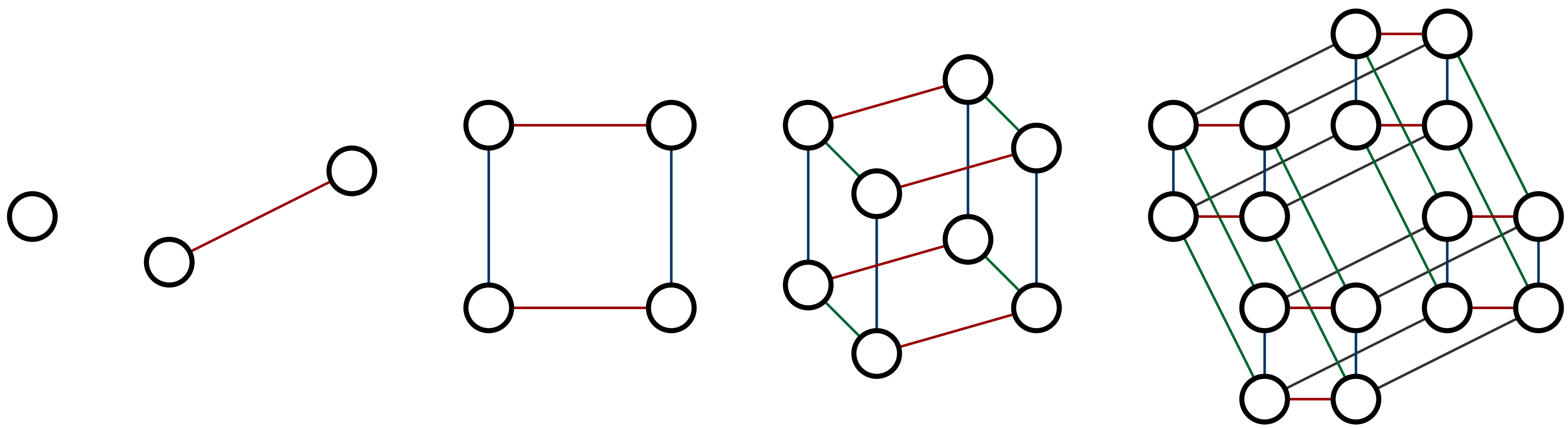
David Eppstein

Univ. of California, Irvine
Donald Bren School of Information and Computer Sciences

Hypercubes

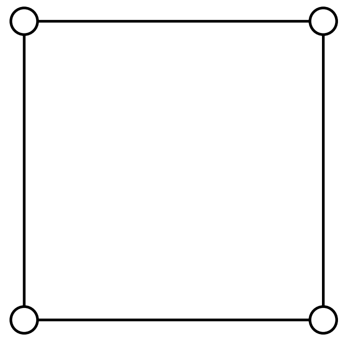
Cartesian product of unit intervals
= convex hull of points with all coordinates zero or one

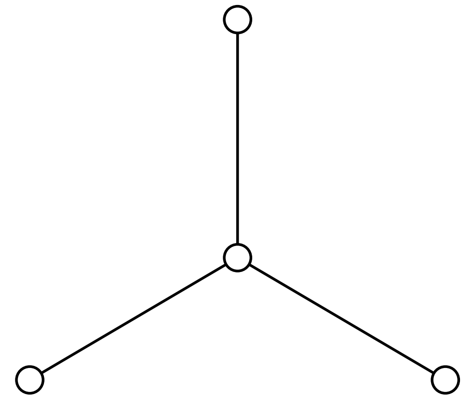
Hamming distance (L_1 metric) between vertices
= number of coordinates at which they differ



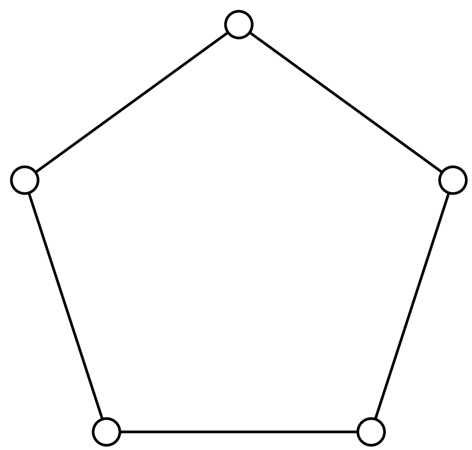
Partial cube = graph with distance-preserving embedding onto a hypercube

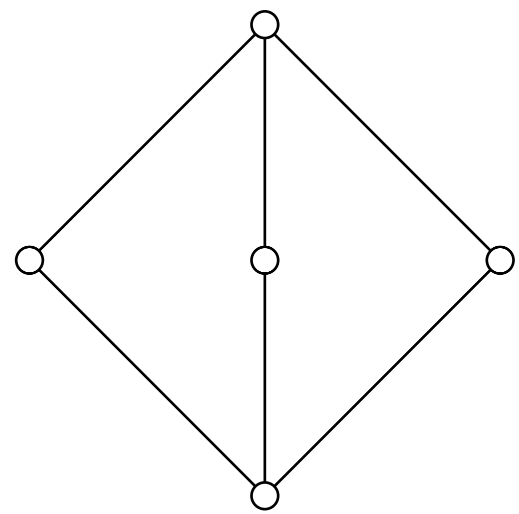
Partial cubes:





Not partial cubes:





Medium = state-transition system with partial cube connectivity
(used to model voter states in political choice theory)

Examples of partial cubes

Arrangement of hyperplanes

vertices = arrangement cells
edges = adjacency between cells

Topological orderings of DAGs

vertices = topological orderings
edges = swap vertices adjacent in order
but nonadjacent in graph

Acyclic orientations of undirected graph

vertices = orientations
edges = reverse orientation of single edge

Weak orderings (total ordering with ties) of n-item set

vertices = orderings
edges = split or merge group of tied items

New results:

Two drawing algorithms for partial cube graphs & media

Embed into low-dim lattice then project onto the plane

- Polynomial time
- Works for any partial cube
- Drawing has crossings but good separation properties

Planar drawing with all faces = symmetric polygons

- Linear time
- Graphs with such drawings = subset of partial cubes

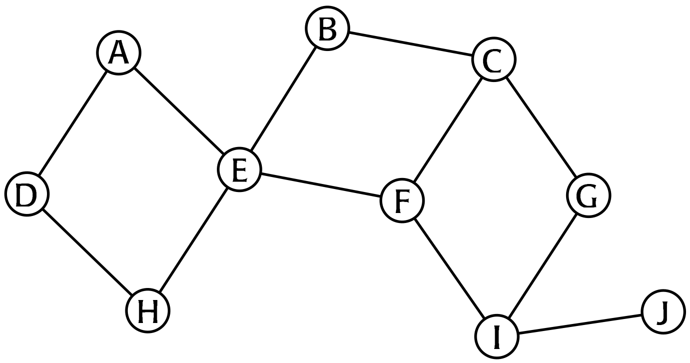
Minimum-dimension lattice embedding

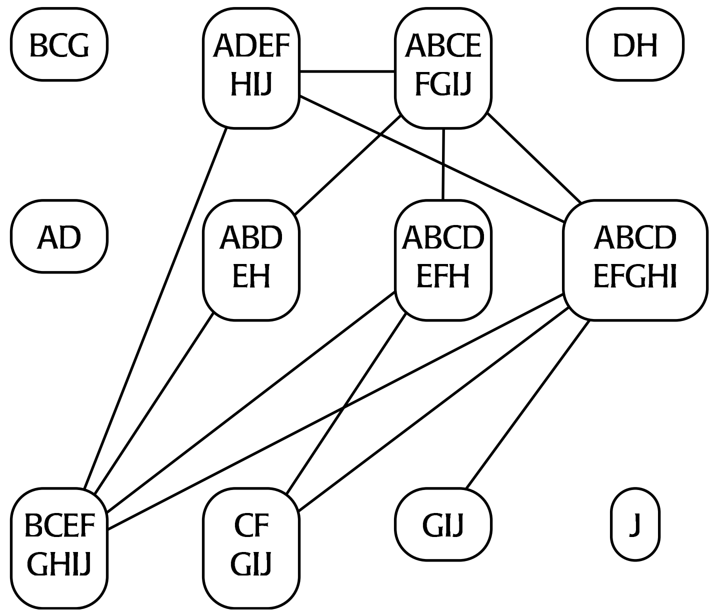
[Eppstein, Eur. J. Combinatorics, to appear]

Define ***semicube graph*** of partial cube

vertices = halves of underlying hypercube

edge = two halves cover partial cube with nonempty intersection





Min dimension of integer lattice containing the partial cube
= hypercube dimension - # edges in maximum matching of semicube graph

From lattice embeddings to drawings

If min dimension = 2

Already have a planar drawing with square faces

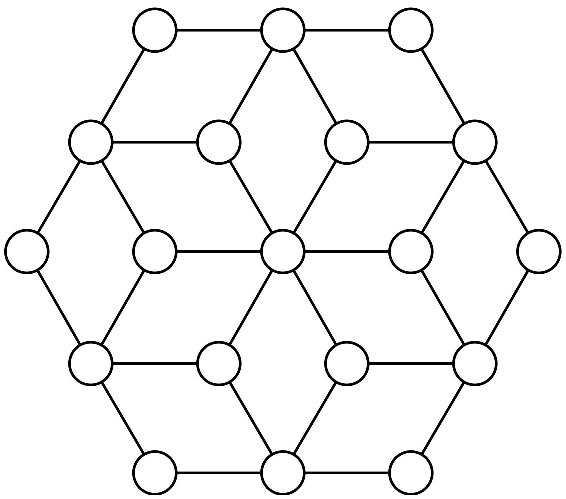
Unit-distance vertex separation, right angle separation

If min dimension = 3

Search for main-diagonal projection distinguishing all vertices from each other

Gives planar drawing with 60-120 rhombus faces

Good vertex and angle separation



From lattice embeddings to drawings

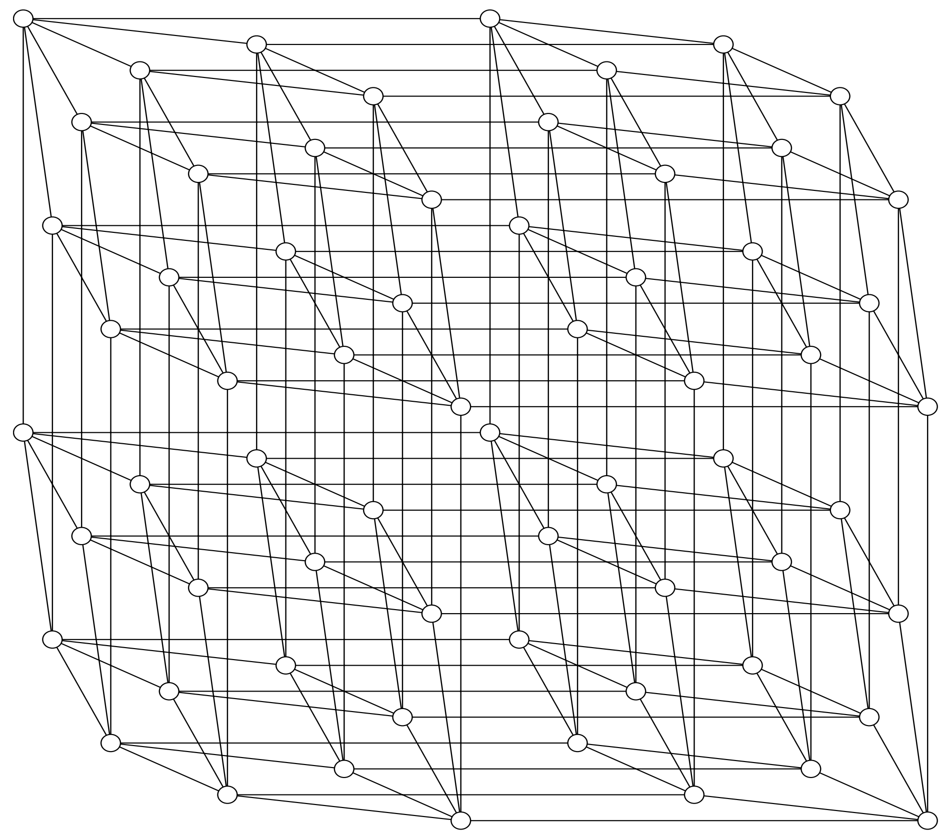
If min dimension > 3 or no 3d planar projection exists

For each dimension i of lattice
choose integer X_i, Y_i

Values X_i chosen in order by i
so that
sets of vertices with same coords ≥ i
project to distance ≥ 1 apart

Values Y_i chosen similarly
but in reverse order by i
separating vertices with same coords ≤ i

Project lattice point v to $(X \bullet v, Y \bullet v)$



Properties of projected lattice drawings

All vertices have distinct integer positions

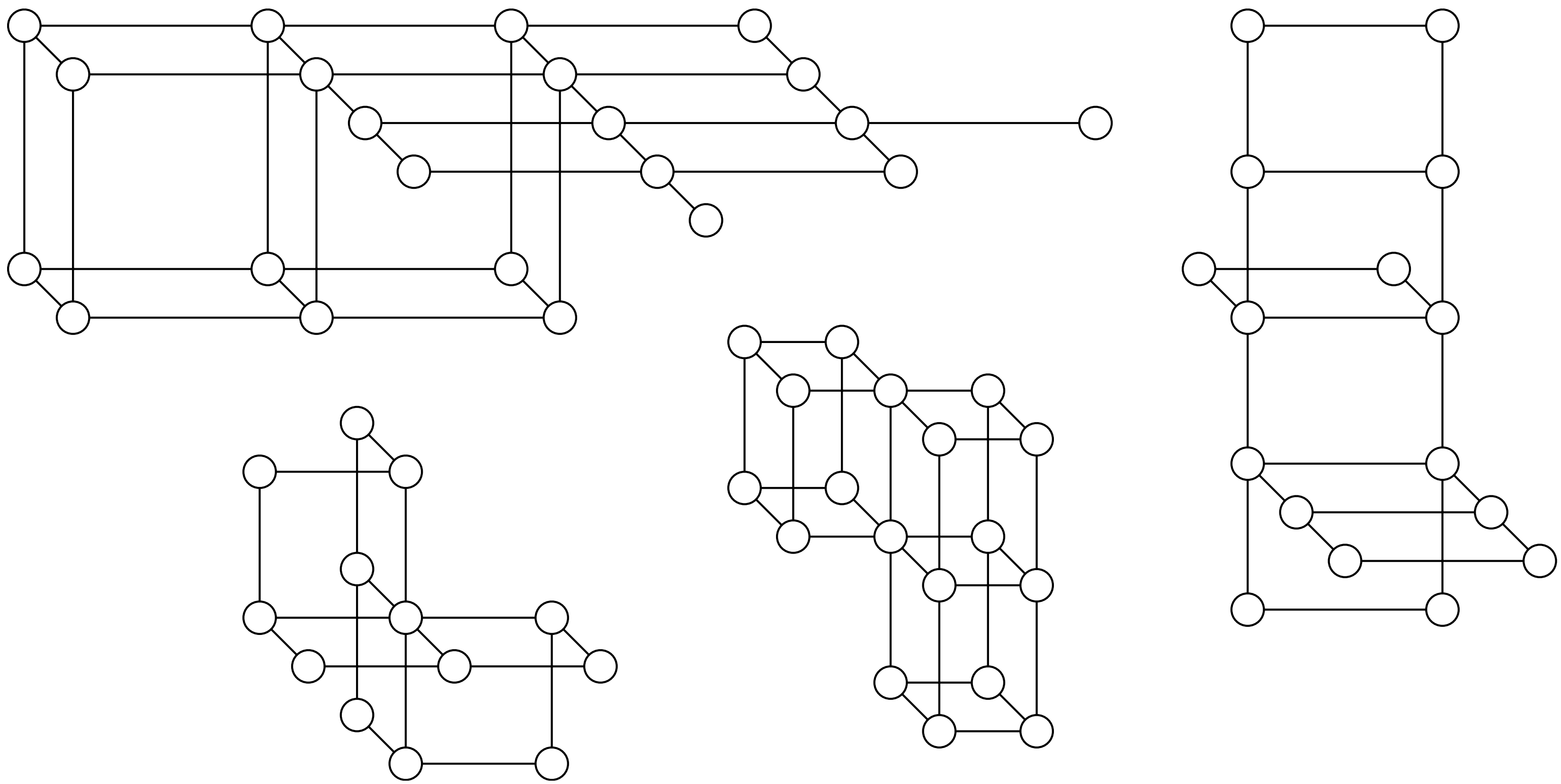
All edges are drawn as straight line segments

Unit separation between vertices and nonadjacent edges

Edges drawn parallel iff they are parallel in lattice
(so lattice structure easy to recover from drawing)

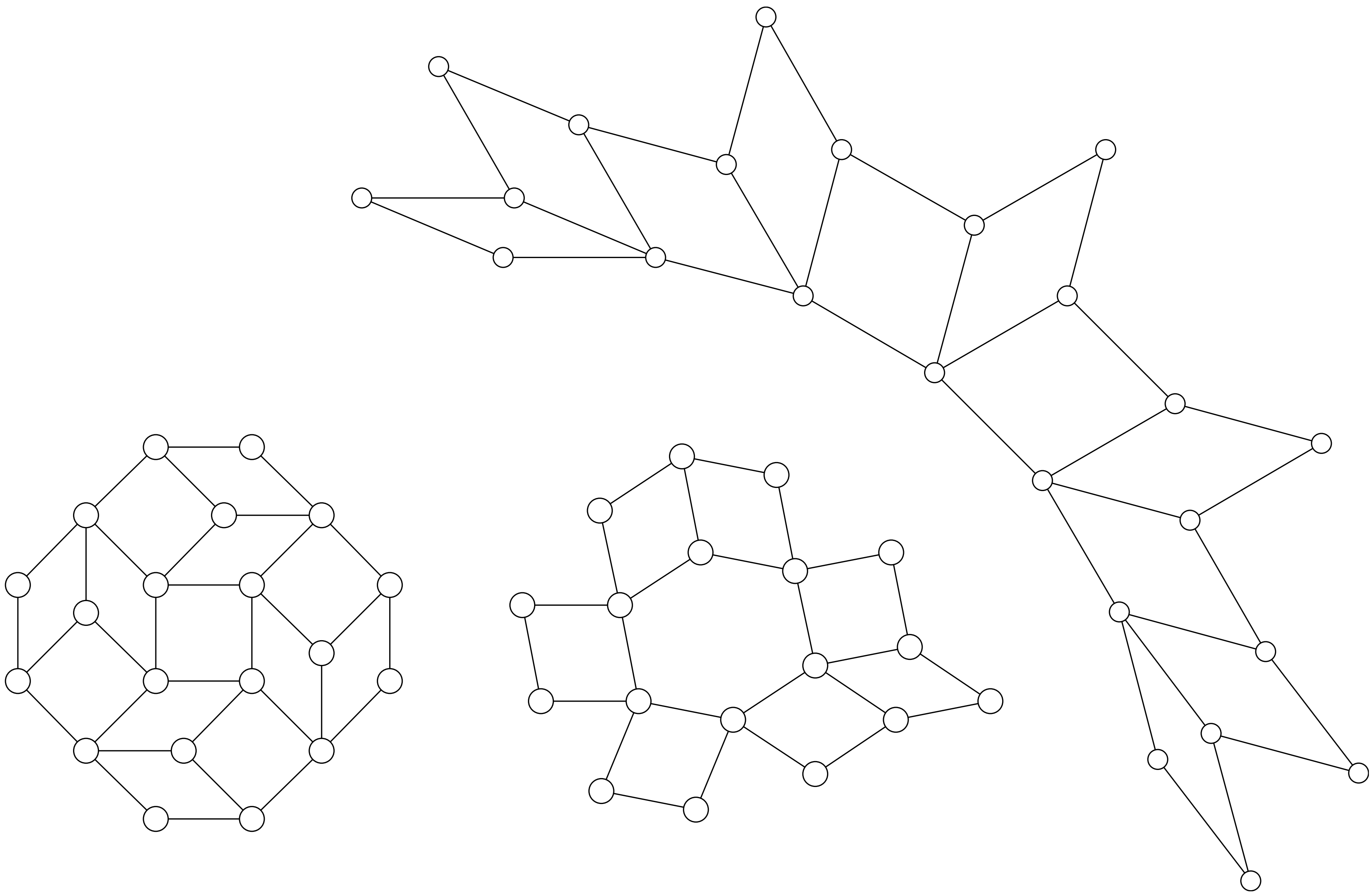
Quadratic area bound for drawings of hypercubes
(but not necessarily for partial cubes)

More examples of projected lattice drawings



Face-symmetric planar drawings

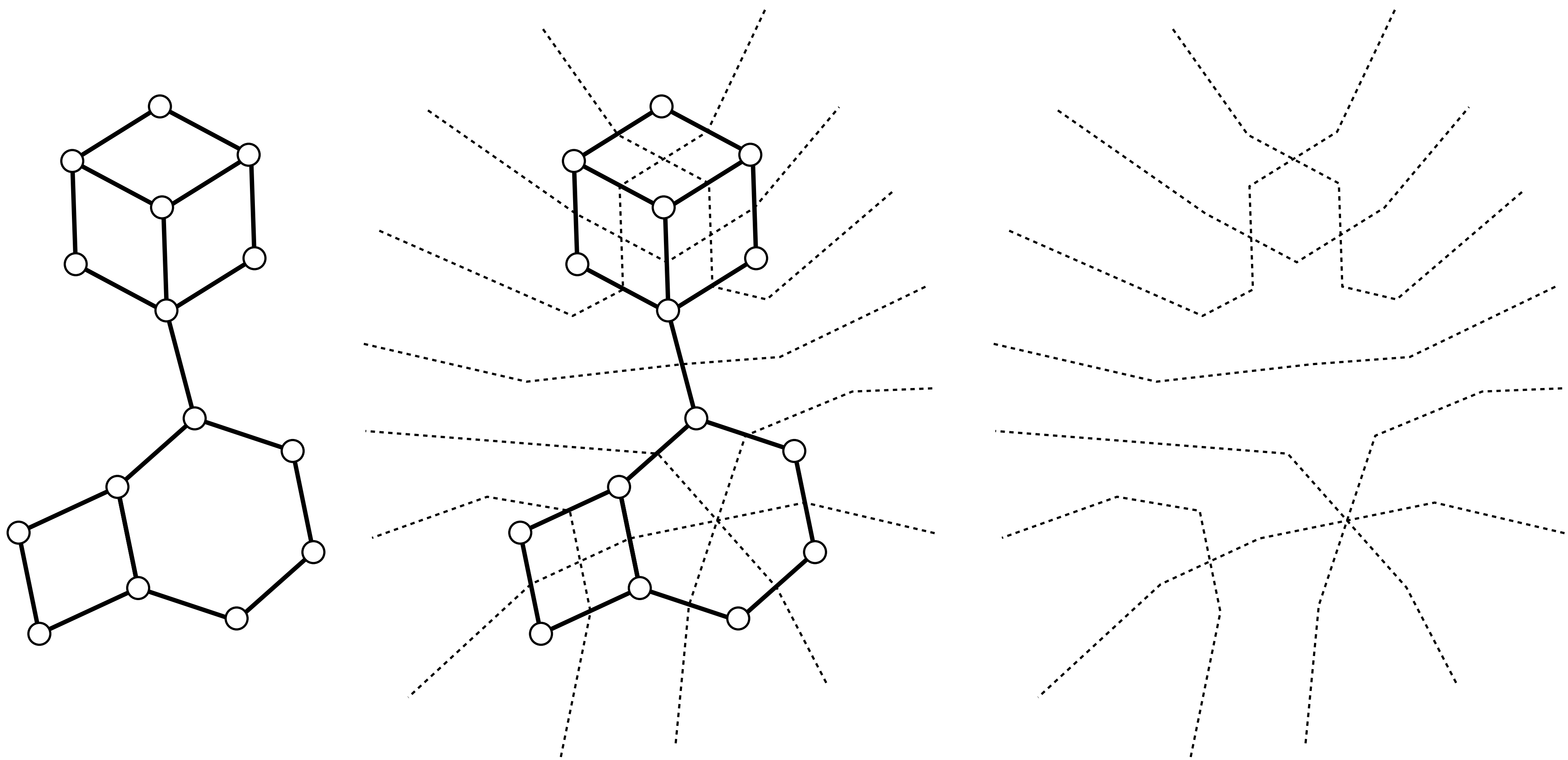
Planar straight line graph drawing
All faces (except the outer one) are centrally symmetric convex polygons



Graphs with face-symmetric drawings are dual to weak pseudoline arrangements

Arrangement of curves, cross at most once per pair, endpoints in outer face

Form by connecting midpoints of opposite edges in each face

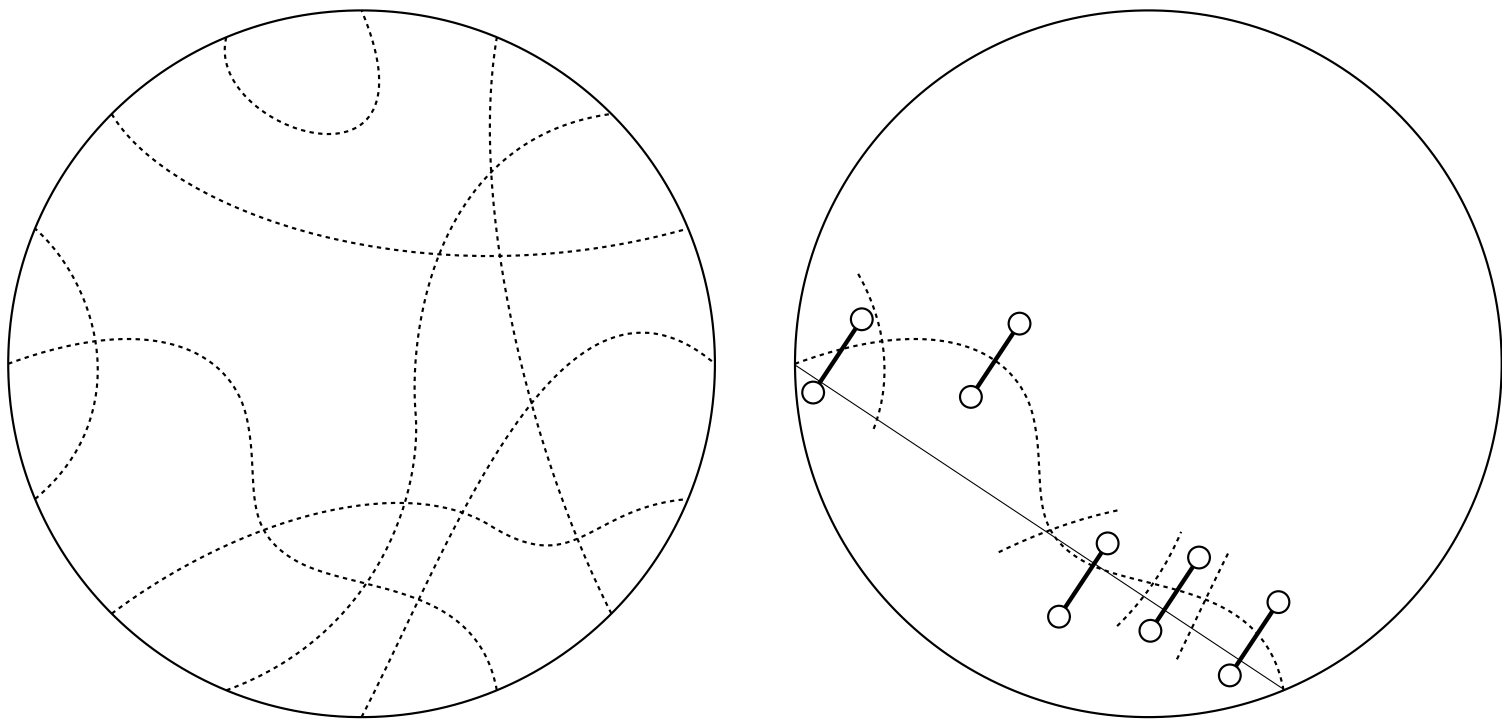


Therefore, all such graphs are partial cubes

Weak pseudoline arrangements are dual to graphs with face-symmetric drawings

Choose vector for each pseudoline so no >180-degree concavities

Space curve endpoints equally around large circle
Choose unit vector perpendicular to segment between curve endpoints



Apply lattice projection method
to hypercube embedding with dimension = # curves in arrangement

Finding a face-symmetric drawing

Find the dual weak pseudoline arrangement

Linear-time algorithm: based on SPQR tree

Implemented (quadratic) algorithm:
construct arrangement incrementally
(one curve per dimension of hypercube embedding)

Choose vectors and apply vector projection

Result: **can find drawing (if it exists) in O(n) time**

Conclusions

Interesting special class of graphs

Two new drawing algorithms

Drawings highlight the partial cube structure of the graph

Projection method: can read lattice embedding from drawing

Face-symmetric: dual to weak pseudoline arrangement, must be partial cube

Open problems

How to test for existence of a planar lattice projection?
(may not be face-symmetric)

Optimize choice of vectors for face-symmetric drawing?
(e.g. to maximize angular separation)

Better area requirements for either kind of drawing?