Dual graph of a given Planar graph

Biki Chaudhary
Email:ism2014001@iiita.ac.in
Tara Prasad Tripathy
Email:ihm2014003@iiita.ac.in

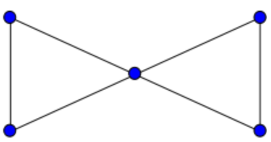
Abstract - The objective of this paper is to find the dual graph from the given planar graph. The main task in finding the dual graph is to find the faces in the planar embedding of a graph. To our knowledge, there are only few easily accessible algorithms to find faces in planar graphs, e.g., the planar face traversal of the Boost Graph Library. We make use of geometrical relationships between vertices and edges to detect the faces in the planar embedding. The general idea of our algorithm is to iterate over all vertices of the graph and enumerate all faces that contain this vertex. From each vertex, the edges are traversed in a certain orientation (i.e., clockwise or counterclockwise) in order to find all faces that contain the initial vertex.

Index Terms - Planar graph, Non-planar graph, Dual graph, Planar embedding, Complete graph, Faces, Orientation

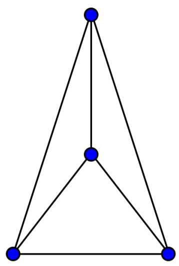
I. INTRODUCTION

In graph theory, a planar graph is a graph that can be embedded in the plane i.e., it can be drawn on the plane in such a way that its edges intersect only at their endpoints. In other words, it can be drawn in such a way that no edges cross each other. Such a drawing is called a plane graph or planar embedding of the graph. Some examples of planar graph and non-planar graphs are:

Planar Graphs

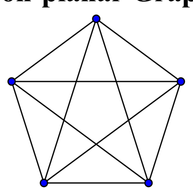


Butterfly Graph

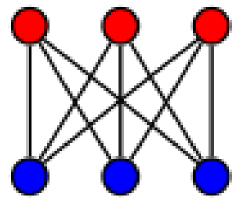


Complete Graph(K_4)

Non-planar Graphs



Complete Graph(K_5)



Utility Graph$_{3,3}$

For a simple, connected, planar graph with v vertices and e edges and f faces, the following simple conditions hold for $v \geq 3$:

- Theorem 1. $e \leq 3v - 6$.
- Theorem 2. If there are no cycles of length 3, then $e \leq 2v - 4$.
- Theorem 3. $f \leq 2v - 4$.

The dual graph of a plane graph G is a graph that has a vertex for each face of G. The dual graph has an edge whenever two faces of G are separated from each other by an edge, and a self-loop when the same face appears on both sides of an edge. Thus, each edge e of G has a corresponding dual edge, whose endpoints are the dual vertices corresponding to the faces on either side of e. The definition of the dual depends on the choice of embedding of the graph G, so it is a property of plane graphs (graphs that are already embedded in the plane) rather than planar graphs (graphs that may be embedded but for which the embedding is not yet known). For planar graphs generally, there may be multiple dual graphs, depending on the choice of planar embedding of the graph.

We cannot compute the dual of a *planar* graph, but only the dual of a *plane* graph. So, for our problem we assume that the planar embedding of the graph is available, i.e., the plane graph. The main task is to find the faces in the plane of the graph. We find the planes from the given plane graph as follows: Firstly, we choose one vertex in each face of *G* (including the outer face) and for each edge *e* in *G* we introduce a new edge in *G** connecting the two vertices in *G** corresponding to the two faces in *G* that meet at *e*. Furthermore, this edge is drawn so that it crosses *e* exactly once and that no other edge of *G* or *G** is intersected. Then *G** is again the embedding of a (not necessarily simple) planar graph; it has as many edges as *G*, as many vertices as *G* has faces and as many faces as *G* has vertices. An example of planar graph(blue) with its dual graph(red):

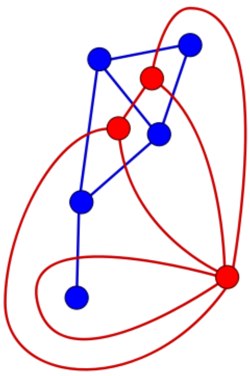


Figure1: Planar Graph with its Dual graph

II. MOTIVATION

Testing a graph for planarity and embedding the planar graph in the plane have several applications. For example, the design of integrated circuits and the layout of printed circuit boards require testing whether a circuit can be embedded in the plane without edge crossings. Several cases of the routing problem have been shown to be equivalent to constructing planar embedding of special classes of graphs. Determining isomorphism of chemical structures is simplified if the structures are known to be planar. Because of the great practical interest, these two problems- planarity testing and planar embedding have been extensively studied in the literature.

The duality of the planar graphs has applications in several other areas of mathematical and computational study. In geographic information systems, flow networks are dual to cellular networks describing drainage divides. In computer vision, digital images are partitioned into small square pixels, each of which has its own color. The dual graph of this subdivision into squares has a vertex per pixel and an edge between pairs of pixels that share an edge. It is useful for applications including clustering of pixels into connected regions of similar colors. Because of the great practical interest, planar graphs and dual graphs are highly studied in the literature.

III. ALGORITHMS

The general idea of our algorithm is to iterate over all vertices of the graph and enumerate all faces that contain this vertex. From each vertex, the edges are traversed in a certain orientation (i.e., clockwise or counterclockwise) in order to find all faces that contain the initial vertex. To determine the edges we need to traverse, we compute the orientation and the angle between the current edge and the next candidate edge, as described next.

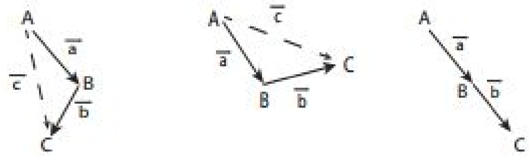


Figure 1: Clockwise (left), counterclockwise (middle), and collinear (right) orientation of two vectors a and b. The dashed line is the additional vector used to compute the orientation of a and b.

Orientation

The orientation is easily computed using the cross product between the incoming edge (vector a, Fig. 1) and the vector c (Fig. 1) between the starting vertex (A) and the ending vertex (C):

$$\vec{a} \times \vec{c} = \begin{pmatrix} a_x & c_x \\ a_y & c_y \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ a_x c_y - a_y c_x \end{pmatrix}.$$

In case of a clockwise orientation of a and b, the cross product is positive, whereas it is negative for a counterclockwise orientation. If more than one candidate edge are in counterclockwise orientation, we take the one with the smallest angle. If there are only clockwise edges, we take the one with the largest angle.

Angle

If there is more than one incident edge of the same orientation, we compute the angles between the current edge and all incident edges:

$$\alpha_{rad} = \arccos\left(\frac{\vec{a}\cdot\vec{b}}{|\vec{a}|\cdot|\vec{b}|}\right),$$

$$\alpha_{deg} = \frac{\alpha_{rad}\cdot 180}{\pi}.$$

The overall algorithm is summarized in Alg. 1.

Algorithm 1: To find faces from plane graph G

for all v ∈ V do
  for all adjacent v_{adj} to v **do** initialize visit
    while v ∉ visit **do**
      if |visit| = 3 and counterclockwise **then**
        for all candidate vertices v_{cand} **do**
          if $|v_{cand}|$ = 1 **then** add v_{cand} to visit
          else
            calculate orientation o and angle α
            find v_{cand} leading to counterclockwise orientation v_{ccw}
              if $|v_{ccw}|$ = 1 **then** add v_{ccw} to visit
              else if $|v_{ccw}|$ > 1 **then** add v_{cand} with minimal α to visit
              else if $|v_{ccw}|$ = 0 **then**
                if any v_{cand} is collinear **then**
                  add e_{cand} to visit
                else
                  add v_{cand} with maximal αto visit

Algorithm 2: Construction of the dual graph of G

Input: Set of faces F, where each face is represented as a set of edges

Output: The dual graph G'

Let n = number of faces in F

for i = 1 to n
  for j = i + 1 to n
    if set_intersection(F[i], F[j]) > 0
      add edge(i, j) to G'
return G'

IV. IMPLEMENTATION AND RESULTS

The runtime of algorithm 1 depends on |V| and on the topology of the graph. In the worst-case where the graph consists of a single face, the time complexity would be $O(|V|^2)$. If the number of faces is F, then Algorithm 2 takes $O(F^2)$ as it considers each pair of faces to decide whether an edge is to be added or not. The space complexity is dominated by the number of elements in the adjacency matrix, $O(|V|^2)$.

V. CONCLUSION

Our problem was to find the dual of a plane graph. The main task is to find the faces in the plane of the graph. The present algorithm correctly finds all faces in a planar embedding of a graph as long as all faces are simple polygons. A simple polygon encloses an area of finite size, has exactly two edges meet at each vertex, and has a total number of vertices that is equal to the total number of edges. The time complexity is quadratic in terms of the number of vertices.

VI. REFERENCES

1. http://mosaic.mpi-cbg.de/docs/Schneider2015.pdf
2. https://math.stackexchange.com/questions/8140/find-all-cycles-faces-in-a-graph
3. https://math.stackexchange.com/questions/468117/algorithm-for-creating-the-dual-graph-of-a-given-planar-graph
4. https://en.wikipedia.org/wiki/Dual_graph
5. https://en.wikipedia.org/wiki/Planar_graph