GRAPH THEORY COMPENDIUM

Michele Zito

September 16, 2005

In this chapter I will describe a number of beautiful algorithmic results in Graph Theory. The interested reader is referred to the following texts:

- Harary (1969) *Graph Theory*.
- Gibbons (1985) *Algorithmic Graph Theory*.
- Deistel (1997) *Graph Theory* (available on-line from the author's web page).

1 Preliminaries

Most of the graph-theoretic terminology will be taken from [3]. A (*simple undirected*) *graph* $G = (V, E)$ is a pair consisting of a finite nonempty set $V = V(G)$ of *vertices* (or *nodes* or *points*) and a collection $E = E(G)$ of distinct subsets of V each consisting of two elements called *edges* (or *lines*). If $e = \{u, v\} \in E$ then the vertices u and v are *adjacent*, vertex u and the whole edge e are *incident* (or else we say that u belongs to e, sometimes using the set-theoretic notation $u \in e$). Also if $f = \{v, w\} \in E$ then e and f are incident. If $F \subseteq E(G)$ then $V(F)$ is the set of vertices incident to some $e \in F$. For every $U \subseteq V(G)$, $N(U)$ will denote the set of vertices adjacent to some $v \in U$ and not belonging to U. If $U = \{v\}$ we write $N(v)$ instead of $N(\{v\})$. If $U, W \subseteq V$ then $\mathrm{cut}(U, V)$ is the set of edges having one endpoint in U and the other in W.

The *degree* of a vertex v is defined as $\deg_G v =_{df} |N(v)|$. The *minimum* (resp. *maximum*) *degree* of G is $\delta = \delta(G) = \min_{v \in V} \deg_G v$ (resp. $\Delta = \Delta(G) = \max_{v \in V} \deg_G v$). For all $i \in \{0, \ldots, n-1\}$ let $V_i(G) = \{v \in V : \deg_G v = i\}$. A *multiset* is a collection of objects in which a single object can appear several time. A *multigraph* is a pair $H = (U, E)$ in which U is the set of vertices and E is a multiset of edges. If e appears $x_e > 1$ times in E then each of its occurrences is a *parallel edge*. The *skeleton* of a multigraph $H = (U, E)$ is a graph G with $V(G) = U$ and $E(G)$ containing a single copy of every parallel edge in H plus all the $e \in E$ with $x_e = 1$. A graph is *directed* if the edges are ordered pairs. Round brackets will enclose vertices belonging to a directed edge.

A graph is *labelled* if its vertices are distinguished from one another by names. Figure 1 shows the 64 different labelled graphs on four vertices. Some of these graphs only differ for the labelling of their vertices, their topological structure is the same. More formally, two graphs G_1 and G_2 are *isomorphic* if there is a one-to-one correspondence between their labels which preserves adjacencies. A graph is *unlabelled* if it is considered disregarding all possible labelling of its vertices that preserve adjacencies. Figure 2 shows the eleven unlabelled graphs on four vertices.

A graph is completely determined by either its adjacencies or its incidences. This information can be conveniently stated in matrix form. The *adjacency matrix* of a labelled undirected (resp. directed) graph $G = (V, E)$ with n vertices, is an $n \times n$ matrix A such that, for all $v_i, v_j \in V$, $A_{i,j} = 1$ if v_i is adjacent to v_j (resp. if $(v_i, v_j) \in E$) and $A_{i,j} = 0$ otherwise.

A *subgraph* of $G = (V, E)$ is a graph $H = (W, F)$ with $W \subseteq V$ and $F \subseteq E$. H is a *spanning subgraph* if $W = V$ and it is an *induced subgraph* if whenever $u, v \in W$ with $\{u, v\} \in E$ then

Figure 1: The 64 distinct labelled graphs on 4 vertices.

$\{u, v\} \in F$. If $W \subseteq V(G)$ we will denote by $G[W]$ the induced subgraph of G with vertex set W. K_n is the *complete* simple graph on n vertices. It has $n(n-1)/2$ edges. Every graph on n vertices is a subgraph of K_n.

A graph $G = (V, E)$ is *bipartite* if V can be partitioned in two sets V_1 and V_2 such that every line of G joins a vertex in V_1 with a vertex in V_2. K_{n_1,n_2} is the complete bipartite graph on $n = n_1 + n_2$ vertices. A graph is *planar* if it can be drawn on the plane so that no two edges intersect.

If G is a graph and $v \in V$ then $G - v$ is the graph obtained from G by removing v and all edges incident to it; if $v \notin V$ then $G + v = (V \cup v, E)$. If $e = \{u, v\} \in E$ then $G - e = (V, E \setminus \{e\})$ and $G + e = (V \cup \{u, v\}, E \cup e)$. These operations extend naturally to sets of vertices and edges.

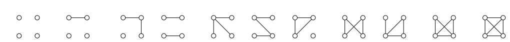

Figure 2: The 11 distinct unlabelled graphs on 4 vertices.

A *path* in a graph $G = (V, E)$ is an ordered sequence of vertices formed by a *starting vertex* v followed by a path whose starting vertex belongs to $N(v)$. The path is *simple* if all vertices in the sequence are distinct. The *length of a path* $P = (v_1, \ldots, v_k)$ is $k - 1$. A *cycle* is a simple path $P = (v_1, \ldots, v_k)$ such that $v_1 = v_k$. A single vertex is a cycle of length zero. Since $v \notin N(v)$ there is no cycle of length one. An edge $\{u, v\} \in E$ belongs to a path $P = (v_1, \ldots, v_k)$ if there exists $i \in \{1, \ldots, k-1\}$ such that $\{u, v\} = \{v_i, v_{i+1}\}$. Two vertices u and v in a graph are *connected* if there is a path $P = (v_1, \ldots, v_k)$ such that $\{u, v\} = \{v_1, v_k\}$. The *distance* $dst_G(u, v)$ between them is the length of a shortest path between them. The subscript G will be omitted when clear from the context. A *connected component* is a subgraph whose vertex set is $U \subseteq V$, such that all $u, v \in U$ are connected and no $v \in V \setminus U$ is connected to some $u \in U$. A graph is *connected* if all its vertices are connected. A *tree* is a connected graph containing no cycles. Any graph with no cycles is a *forest*. A tree in which one vertex, the *root*, is distinguished, is called a *rooted-tree*. In a rooted-tree any vertex of degree one, except the root, is called a *leaf*. There is precisely one path between any two vertices of a tree. The *depth* or *level* of a vertex in a rooted-tree is the length of the path from the root to that vertex. If $\{u, v\}$ is an edge of a rooted-tree such that u lies on the path from the root to v, then u is the *father* of v and v is a *child* of u. An ancestor of u is any vertex of the path from u to the root of the tree. Similarly, if u is an ancestor of v, then v is a *descendant* of u. Finally a *binary tree* (resp. k-*ary tree*) s a rooted-tree in which every vertex, unless it is a leaf, has two (resp. k) children.

Graph Theory Applications. At least two:

Shortest tour through a number of cities This is known as the Travelling Salesman Problem. A complete graph is given, with weights on the edges for the distance ... The minimum weight spanning tree of that graph is computed ... we then walk along the spanning tree edges to complete the tour (shortcutting where necessary).

Timetabling Simplified setting:

- All lectures take place on Monday in 4 one hour time slots Let s_i refer to slot $9 + i$ to $10 + i$, for $i \in \{0, \ldots, 3\}$.
- There are 6 modules to be allocated an hour slot (denoted by M_k, for $k = 1, \ldots, 6$), each taught in one hour slot.
- Students have to take 2 modules.

The graph has one vertex for each module and one edge for each pair of modules chosen by a student. The colours are the classes. No edge can have its end-point of the same colour (example with a cubic graph on 6 vertices).

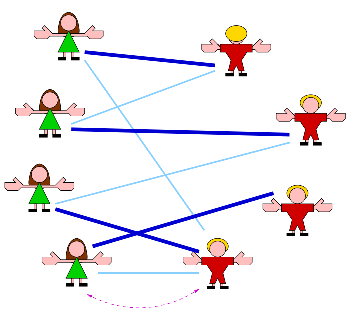

Figure 3: A matching in a graph.

Graph Matchings. If $G = (V, E)$ is a graph, a set $M \subseteq E$ is a *matching* in G if $e_1 \cap e_2 = \emptyset$ for all $e_1, e_2 \in M$. Let $V(M)$ be the set of vertices belonging to edges in the matching. A matching M is *maximal* if for every $e \in E \setminus M$, there exists $f \in M$ such that $e \cap f \neq \emptyset$ (we say that f *covers* e).

A matching M is *induced* if for every edge $e = \{u, v\}$, $e \in M$ if and only if $u, v \in V(M)$ and $e \in E$. A number of parameters can be defined to characterise matchings in graphs:

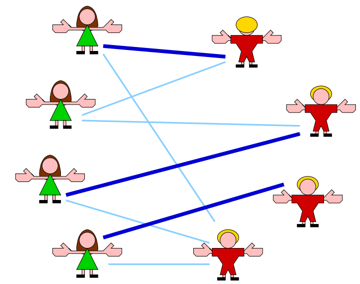

Figure 4: An induced matching in a graph.

Definition 1 *If $G = (V, E)$ is a graph then*

1. *$\beta(G)$ denotes the minimum cardinality of a maximal matching in G;*
2. *$\nu(G)$ denotes the maximum cardinality of a matching in G;*
3. *$\nu_I(G)$ denotes the maximum cardinality of an induced matching in G.*

We will look mainly at $\nu(G)$. The problem of finding a maximum matching in a graph has a glorious history and has an important place among combinatorial problems. The class NP can be characterised as the set of all decision problems for which finding a solution among all possible candidates can take exponential time but checking whether a candidate is a solution only takes polynomial time (see for example [1, Ch. 8]). Maximum matching is a nice example of a problem for which, despite of the existence of an exponential number of candidates, a solution can be found quickly. This fact, discovered by [2], led to a number of algorithmic applications (see for example [4, 5]).

References

[1] J.L. Balcazar, J. Diaz, and J. Gabarro. *Structural Complexity I*, volume 11 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, 1988.

[2] J. Edmonds. Paths, trees and flowers. *Canadian Journal of Mathematics*, 15:449–467, 1965.

[3] F. Harary. *Graph Theory*. Addison-Wesley, 1969.

[4] J. Hopcroft and R. Karp. An $n^{5/2}$ algorithm for maximal matching in bipartite graphs. *SIAM Journal on Computing*, 2:225–231, 1973.

[5] S. Micali and V. V. Vazirani. An $O(v^{1/2}e)$ algorithm for finding maximum matching in general graphs. In *Proceedings of the 21st Annual Symposium on Foundations of Computer Science*, pages 17–27, New York, 1980. IEEE Computer Society Press.