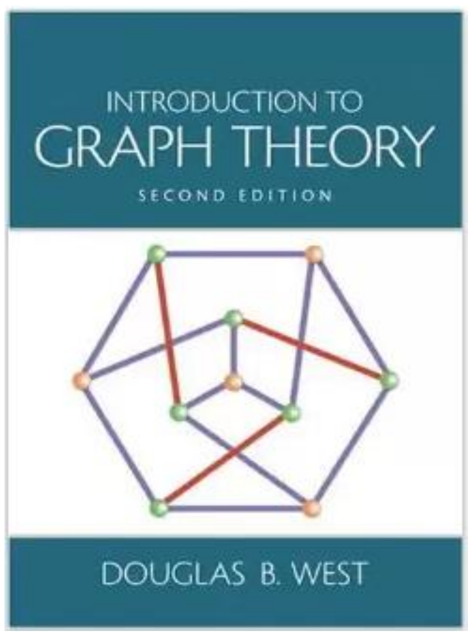
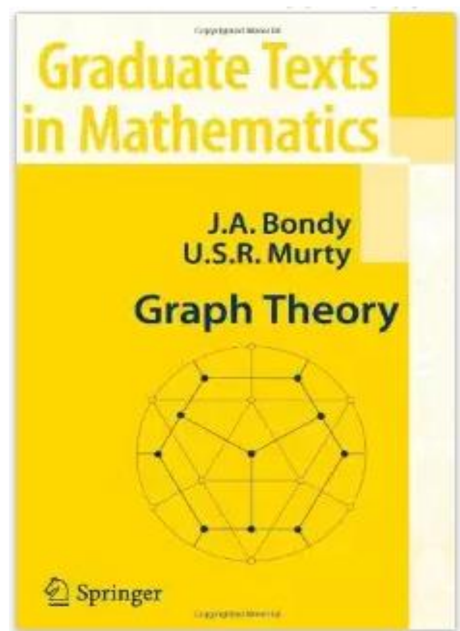
Connected Graphs and Connectivity

prepared and Instructed by

Shmuel Wimer

Eng. Faculty, Bar-Ilan University



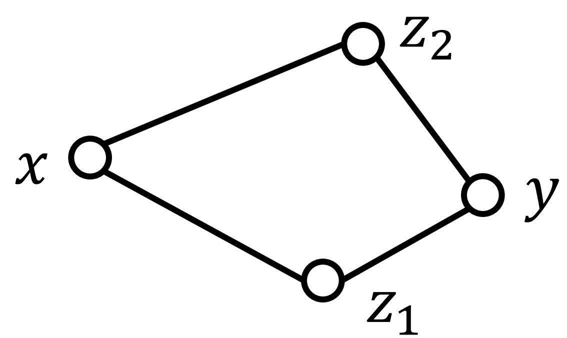


The Friendship Theorem

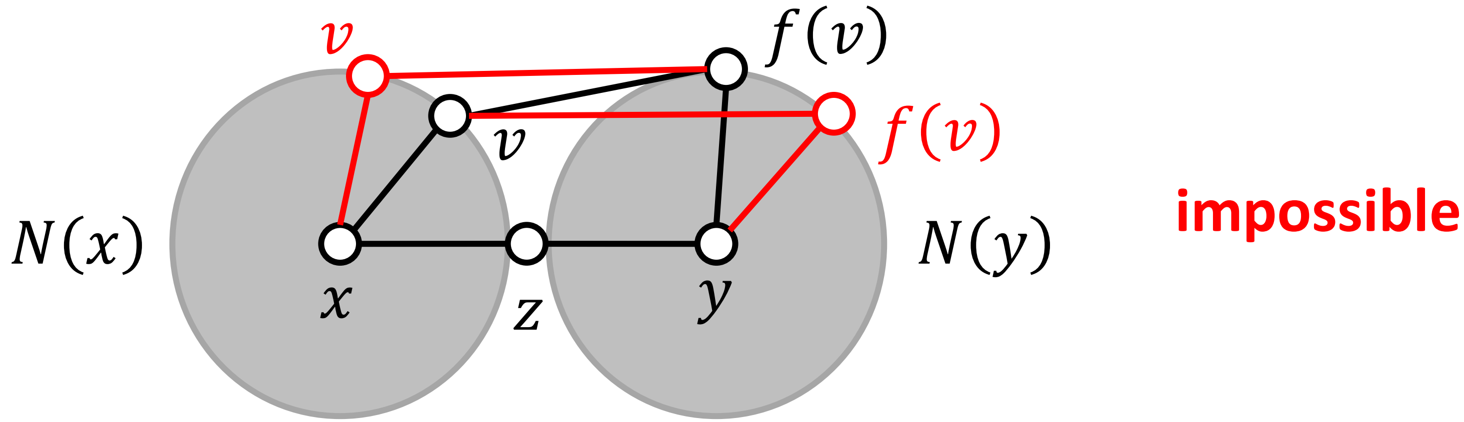
Theorem. (Erdös et. al. 1966) Let G be a simple, n-vertex graph, in which any two vertices (people) have exactly one common neighbor (friends). Then G has a vertex of degree $n-1$ (everyone's friend).

Proof. Suppose in **contrary** that $\Delta(G) < n-1$. We first show that G is regular. Consider two non adjacent vertices x and y, and assume w.l.o.g that $d(x) \geq d(y)$.

Since x and y have a single common neighbor z, G has no C_4 subgraph.



We establish a one-to-one mapping of $N(x)$ to $N(y)$.



For each $v \in N(x) \backslash \{z\}$ denote by $f(v)$ the common neighbor of v and y.

f is therefore one-to-one mapping from $N(x) \backslash \{z\}$ to $N(y) \backslash \{z\}$, hence $|N(x)| = |N(y)|$.

There is $|N(x)| = d(x) \geq d(y) = |N(y)|$, hence $\boldsymbol{d(x) = d(y)}$ for any **non adjacent** vertices of G.

What about z? It could be the friend of all, but by the contrary assumption there is $d(z) < n - 1$.

It has therefore one non adjacent vertex w, and by same one-to-one mapping for z and w $\boldsymbol{d(z) = d(w)}$.

Any other vertex $w \neq z$ is **not a neighbor** of at least one of $\{x, y\}$, say y (G has no C_4). By the same one-to-one mapping for w and y $\boldsymbol{d(w) = d(y)}$.

In conclusion $\boldsymbol{d(x) = d(y) = d(w) = d(z) = k}$. G is therefore $\boldsymbol{k}$**-regular**.

We next look for a relation between n and k by counting the number of 2-edge paths in G.

For $x \in G$ there are $k(k-1)/2$ distinct paths, yielding a total of $\boldsymbol{nk(k-1)/2}$ distinct 2-edge paths.

By the theorem hypothesis, any two vertices have a unique common adjacent.

Picking two vertices yields a total of $\boldsymbol{n(n-1)/2}$ distinct 2-edge paths.

All in all there is $\boldsymbol{n = k^2 - k + 1}$.

To investigate the possible values of k, we examine the **vertex adjacency** matrix $\boldsymbol{A}$ of G.

Since G is k-regular, each row and column of $\boldsymbol{A}$ has k 1s.

Let us consider $\boldsymbol{A}^2$.

Since each row and column of $\boldsymbol{A}$ has k 1s, there is $\boldsymbol{A}^2{}_{ii} = k, 1 \le i \le n$.

Since two vertices have one and only one common neighbor , there is $\boldsymbol{A}^2{}_{ij} = 1, 1 \le i, j \le n, i \ne j$.

$\boldsymbol{A}^2$ is therefore $\boldsymbol{A}^2 = \boldsymbol{J} + (k-1)\boldsymbol{I}$, where $\boldsymbol{J}$ is the $n \times n$ matrix of all 1s and $\boldsymbol{I}$ is the $n \times n$ identity matrix.

the rank of $\boldsymbol{J}$ is 1 so it has an eigenvalue n with multiplicity 1 and eigenvalues 0 with multiplicity $n-1$.

$\boldsymbol{A}^2$ has therefore one eigenvalue $\boldsymbol{n+k-1}$ and $n-1$ eigenvalues $\boldsymbol{k-1}$.

The eigenvalues of $\boldsymbol{A}$ are the square root of $\boldsymbol{A}^2$. By 2-edge path counting argument there is $\boldsymbol{n+k-1=k^2}$.

$\boldsymbol{A}$ has therefore eigenvalues $\pm\sqrt{\boldsymbol{k-1}}$ with multiplicity $n-1$ and eigenvalue $\boldsymbol{k}$ with multiplicity 1.

Since G is simple, $\boldsymbol{A}$'s diagonal entries are all 0, so its trace is 0, and so is the sum of the eigenvalues.

Consequently, there is some integer t (follows from the $\pm$) such that $t\sqrt{k-1}=k$.

The only integers k and t solving $t\sqrt{k-1}=k$ is $\boldsymbol{k=2}$, implying $\boldsymbol{n}=k^2-k+1\boldsymbol{=3}$, hence $\Delta(G)=n-1$, contradicting the supposition $\Delta(G)<n-1$. ■

Euler Tours

A **tour** of a connected graph G is a **closed walk** traversing each edge of a graph at least once.

It is called **Euler tour** if each edge is traversed exactly once. G is **Eulerian** if it admits an Euler tour.

Let G be Eulerian and W an Euler tour with initial and terminal vertex u. Each time an internal vertex $v \in W$ occurs, two edges are accounted.

$d(v)$ is therefore even for all $v \neq u$, and also for u.

Eulerian graph is therefore necessarily **even**.

Lemma. Every maximal trail in an **even** graph is closed.

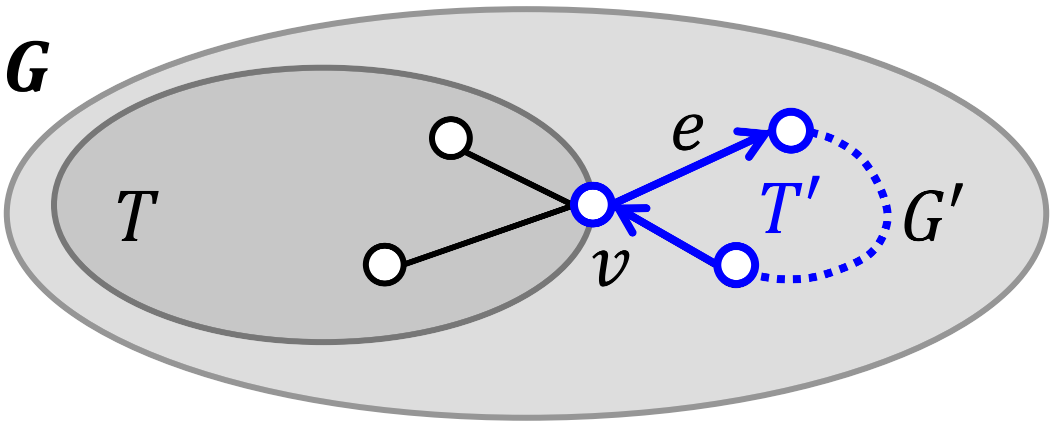
Proof. Let T be a maximal trail but not closed. Since T is open, the terminal edge v has odd incident T's edges.

But then v has another non traversed incident edge which contradicts that T is maximal. ■

Theorem. A finite graph G (parallel edges and loops are allowed) is Eulerian iff it is connected and even.

Proof. Necessity was shown. For sufficiency, let T be a maximal trail in G (must be closed by the lemma).

If $E(T) \neq E(G)$, let $G' = G - E(T)$. G' is even. There must be an edge $e \in E(G')$ incident to T at a vertex v.



Let T' be a maximal trail in G', starting at v along e. By the lemma T' is closed.

Once the traversal of T reached v, we could switch to T', consume its edges and return to v, a contradiction to T being maximal trail at G. ■

Theorem. For a connected nontrivial graph with $2k$ odd vertices (why even?), the minimum number of pairwise edge disjoint trails covering the edges is $\mathbf{max\{1, \boldsymbol{k}\}}$.

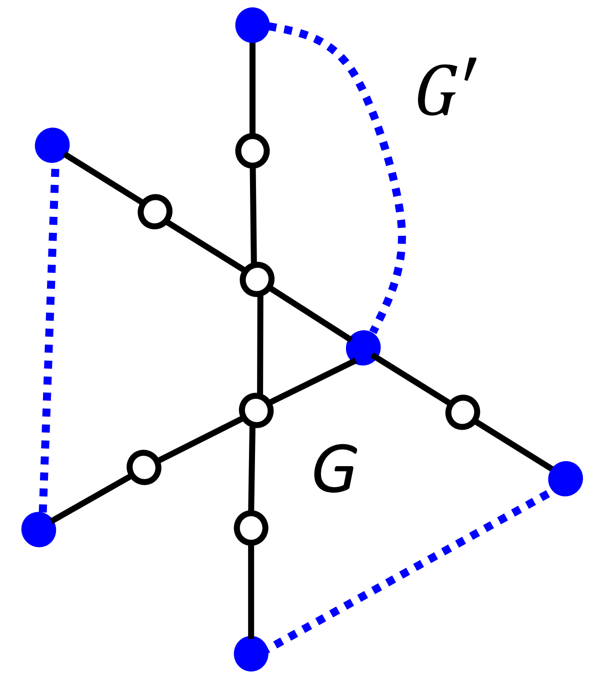
Proof. The internal nodes of trails contribute even degree, and their terminals odd degree.

A trail has two terminals at most (zero if it is a tour) hence k trails at least are required.

One trail at least is required since $|E(G)| > 0$. It was shown also that for $k = 0$ one trail (a tour) suffices, so at least max $\{1, k\}$ is required for $k > 0$.

To see that k edge-disjoint trails cover $E(G)$, pair up the odd vertices of G arbitrarily.

Add an edge connecting each paired odd vertices. G' is connected and each vertex is even, hence an **Euler tour exists**.



Traversing the Euler tour, a new trail starts each time an edge of $E(G') - E(G)$ is traversed, yielding a total of k edge-disjoint trail cover of $E(G)$. ■

Algorithm. (Fleury 1883, Eulerian trail construction)

Input: A connected graph G with at most two odd vertices.

Initialization: Start at an odd vertex if exists, otherwise start arbitrarily at any vertex.

Iteration: Traverse from the current vertex any non cut-edge, unless there is no other alternative.

Theorem. If G has one non trivial component and at most two odd vertices, then Fleury's algorithm constructs an Eulerian trail.

Proof. By induction on $|E(G)|$. Immediate if $|E(G)| = 1$.

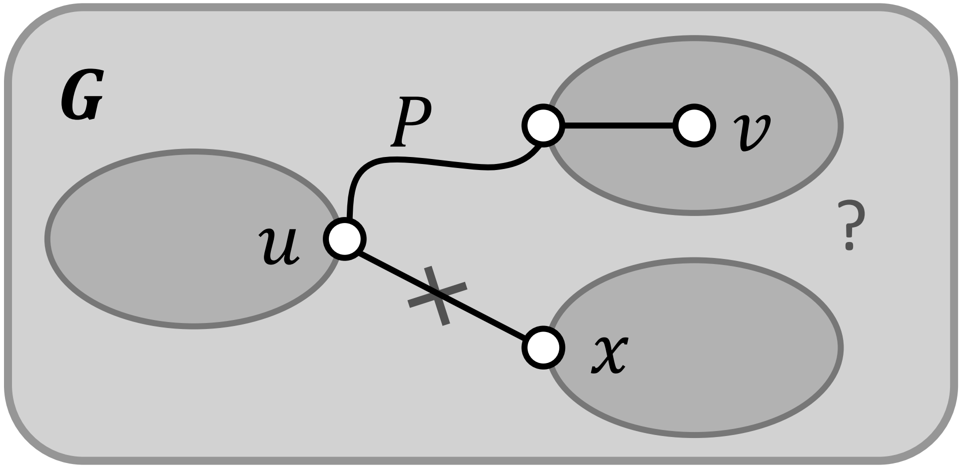
Suppose the construction claim holds for $|E(G)| - 1$.

If G is even, it has no cut-edge. Otherwise, the removal of that edge would leave two separate components, each with a **single odd degree vertex**, which is impossible. (why?)

Consider $G - uv$. $\{u, v\}$ turn to odd degree vertices. Starting from u, by induction Fleury algorithm finds $u \to v$ Eulerian trail. Then close to a tour along vu.

Suppose G has two odd vertices $\{u, v\}$. If $d(u) = 1$, starting from ux, Fleury algorithm finds Eulerian trail from x to v.

So let $d(u) > 1$.



Since G is connected, there is a path P $u \to v$. Since $d(u) > 1$, there is an edge ux not on P.

Assume first $x \neq v$. Remove ux from G. Is $G - ux$ connected?

Yes, otherwise x would have been a single odd vertex of its component, which is impossible. (why?)

$G - ux$ has two odd vertices x and v.

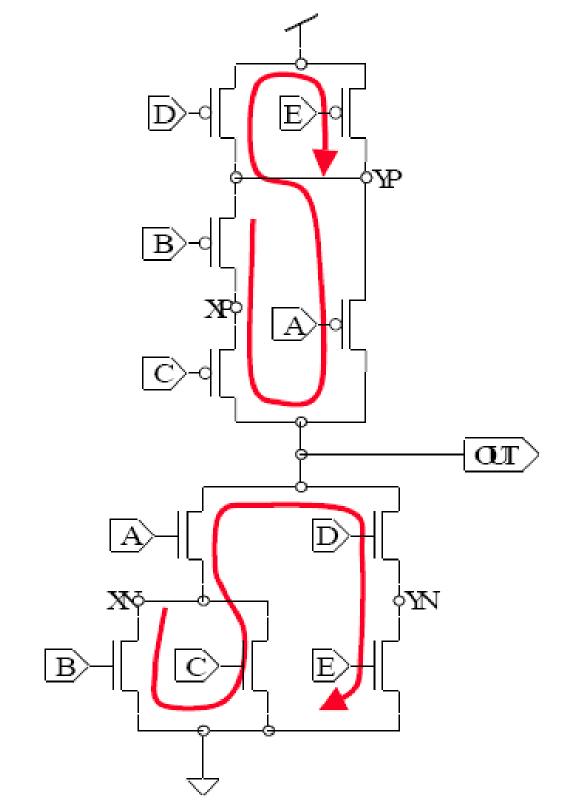
By induction Fleury algorithm finds a $v \rightarrow x$ Eulerian trail, extendable by xu to an Eulerian trail of G.

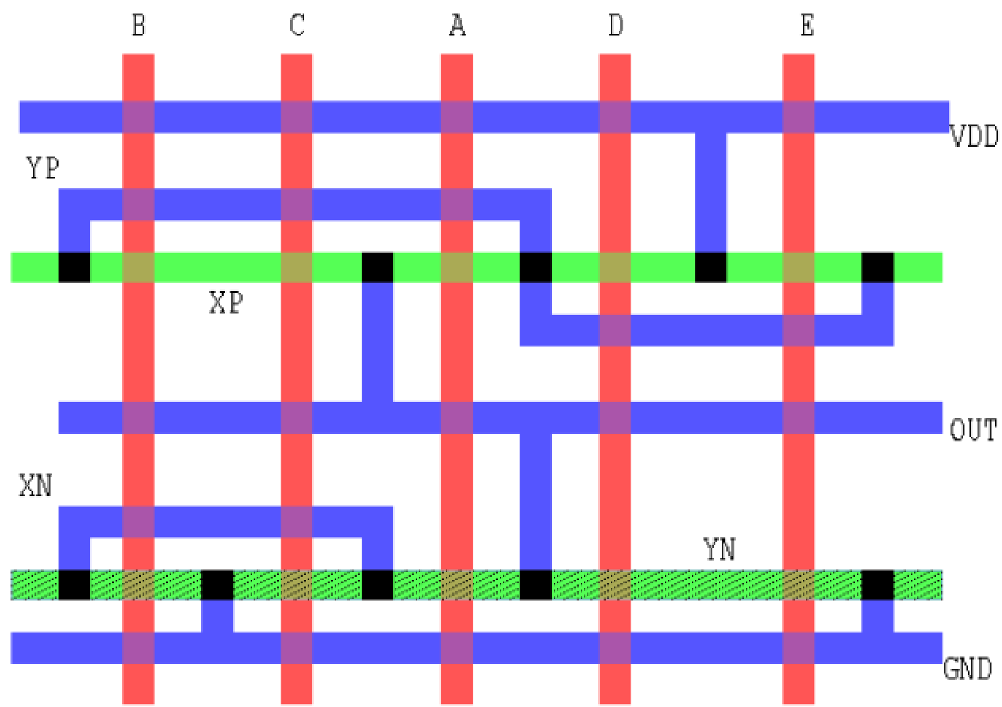
If $x = v$ then $G - uv$ is even.

By induction Fleury algorithm finds Eulerian tour starting (and terminating) at u, extendable by xu to an Eulerian trail of G. ■

Layout of CMOS Compound Gates

$\overline{A\,(B+C) + DE}$





The Chinese Postman Problem

(Guan Meigu 1962) A postman has to traverse all the roads of a town (a graph G), where every road (an edge) has a positive weight (e.g. length, time, biting dogs ☹). The postman starts and ends at the same vertex.

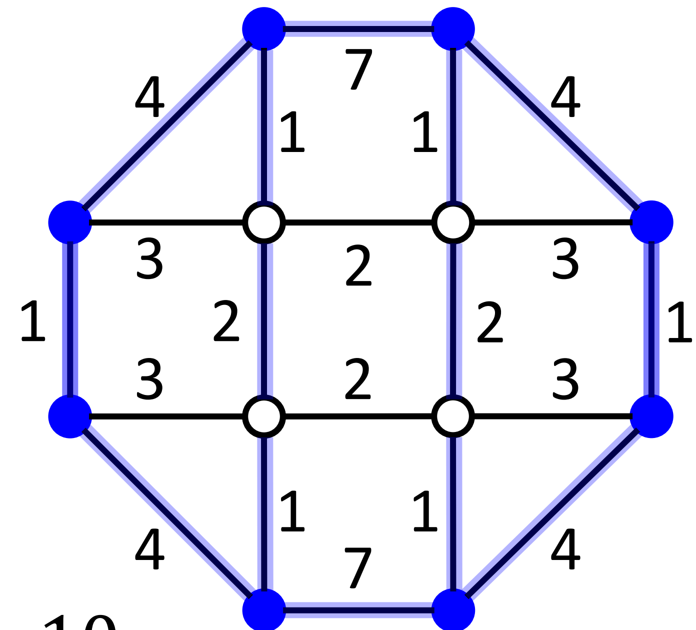
Find a **closed walk of minimum weight** that traverses all the edges.

If G is even, an Eulerian tour is optimal. Otherwise, edges must be repeated (multigarph, parallel edges).

Edges are duplicated to produce an even graph. The problem is therefore to **minimize the total weight** of edge duplicates producing an even graph.

Edges need not be multiplied more than once. (why?)

If an edge is used three or more times, two duplicates can be removed while G stays even.



Cost: $4 + 4 + 4 + 4 = 16.$

Cost: $1 + 7 + 7 + 1 = 16.$

Better solution:
$1 + 2 + 1 + 1 + 2 + 1 + 1 + 1 = 10.$

Duplicated edge connecting odd and even vertices switch their evenness.

Edge addition must proceed until an odd vertex is met.

(Edmonds and Johnson 1973). If there were only two odd vertices, a **shortest path** connecting those solves the problem.

Given $2k$ odd vertices, a weighted K_{2k} graph is defined. An edge weight is the length of the shortest path in *G* connecting the corresponding vertices.

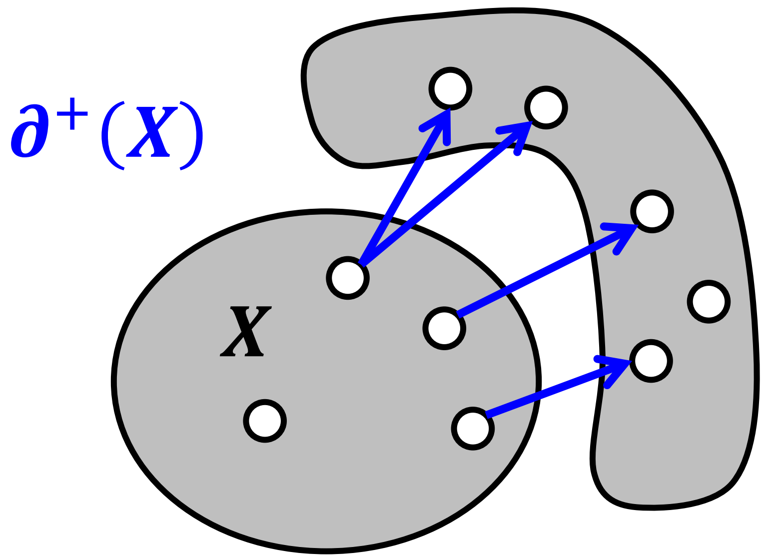
The problem turns into finding a **minimum weight perfect matching** in the above weighted K_{2k}, for which a polynomial algorithm exists. ■

Connection in Digraphs

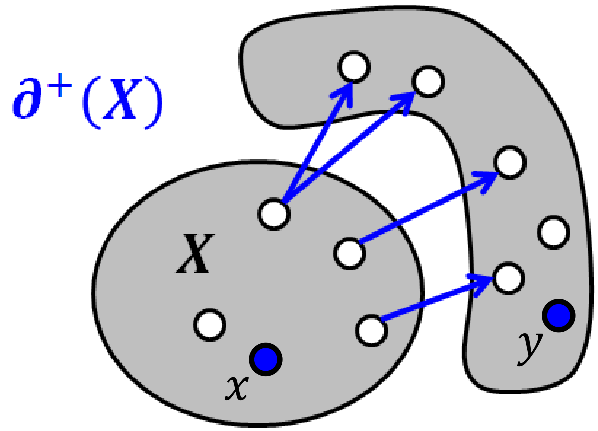
A **directed walk** in a digraph D is an alternating sequence of vertices and arcs $W := (v_0, a_1, v_1, a_2, \dots, v_{l-1}, a_l, v_l)$, such that v_{i-1} and v_i are the tail and head vertices of the arc a_i , respectively, $1 \le i \le l$.

$\boldsymbol{uWv}$ is the portion of W starting at u and ending at v.

$\partial^+(X)$ is the **out-cut** (outgoing arcs) connected to $V(G) \backslash X$.



Theorem. Given digraph G, let $x, y \in V(G)$. y is reachable from x iff $\forall X \subset V(G)$, $x \in X, y \notin X$, there is $\partial^+(X) \neq \emptyset$.



Proof. Let P be a directed path from x to y. Consider any $X \subset V(G)$ such that $x \in X$ and $y \notin X$.

Let y be reachable from x and let $u \in X$ be the last vertex on P and v its successor. Then $uv \in \partial^+(X)$ and hence $\partial^+(X) \neq \emptyset$.

Conversely, suppose that y is not reachable from x, and let X be the set of vertices reachable from x.

There is $y \in V(G) \backslash X$. Since no vertex of $V(G) \backslash X$ is reachable from x, the out-cut $\partial^+(X) = \emptyset$. ■

Definition. A digraph G is **strongly connected** if ∀ **ordered** vertex pair $(u, v) \in V(G)$ there is $u \to v$ path in G.

An Eulerian tour in G implies $d^+(u) = d^-(u) \quad \forall\, u \in V(G)$.

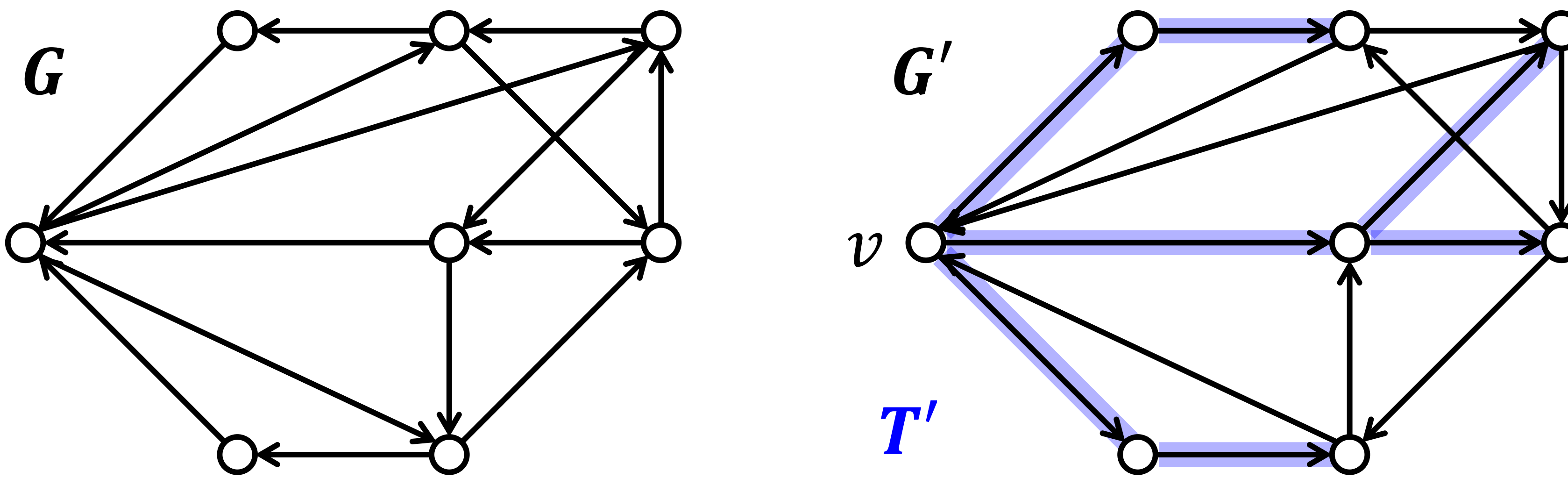
This is also **sufficient**, if the edges of G belong to one connected component. Proofs are similar to undirected graphs.

Algorithm. (Directed Eulerian Tour)

Input. A digraph G that is an orientation of a connected graph, satisfying $d^+(u) = d^-(u)\, \forall\, u \in V(G)$.

Since $d^+(u) = d^-(u)\, \forall\, u \in V(G)$, a $u \to v$ path exists for each $u, v \in V(G)$, hence G is strongly connected (proven later).

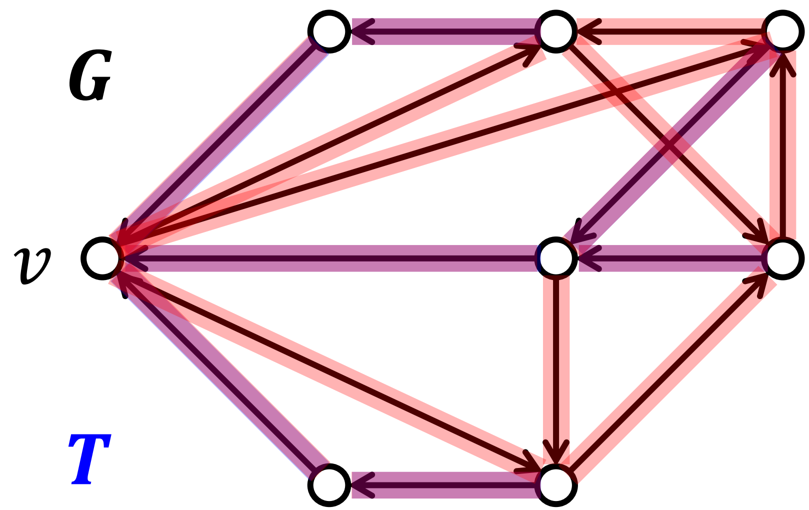
Step 1. Choose a vertex $v \in V(G)$. Derive G' by **reverting** the edge directions of G.



Find a spanning tree T' of G' rooted at v (BFS, other). It is possible since G is strongly connected. Proven in next Theorem.

Step 2. Let T be the reversal of T'. Designate the arc of T.

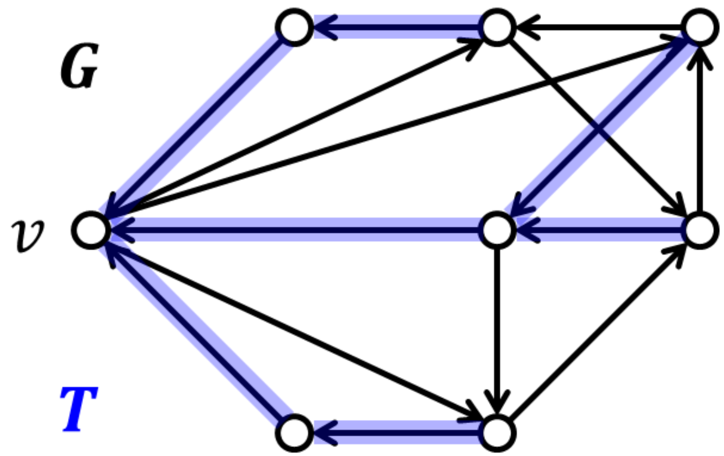
Step 3. Construct an Eulerian tour from v, where leaving from a vertex u is on edge of T only if all other outgoing arcs have already been used. ■



Theorem. If G is multi digraph with one non trivial component and $d^+(u) = d^-(u)\ \forall\, u \in V(G)$, then the algorithm constructs an Eulerian tour of G.

Proof. The construction of T' by BFS (oriented) must reach all $V(G)$. If it did not, let R be the set of those reached and $V(G) \backslash R \neq \emptyset$.

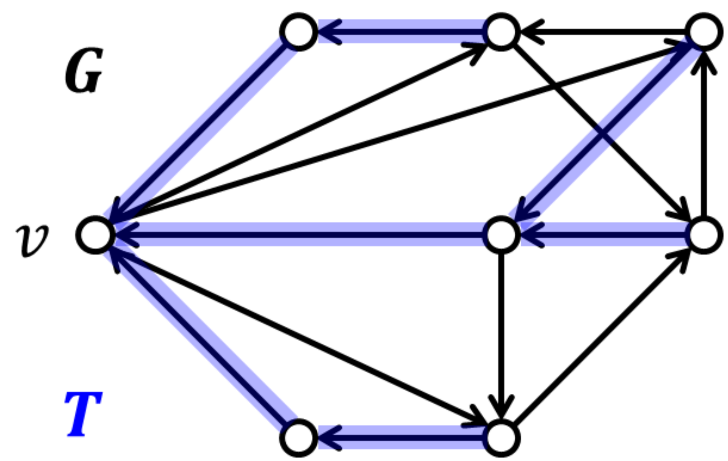
An arc within T' contributes one to the in-degree and one to the out-degree of R's vertices.



R is connected to $G \backslash R$ only by entering arcs, otherwise T' (and R) could be expanded. Such arcs contribute only to the in-degree of R's vertices.

The total in-degree of R's vertices is therefore greater than their total out-degree, which is impossible since $d^+(u) = d^-(u)\ \forall\, u \in V(G)$. Hence $R = V(G)$.

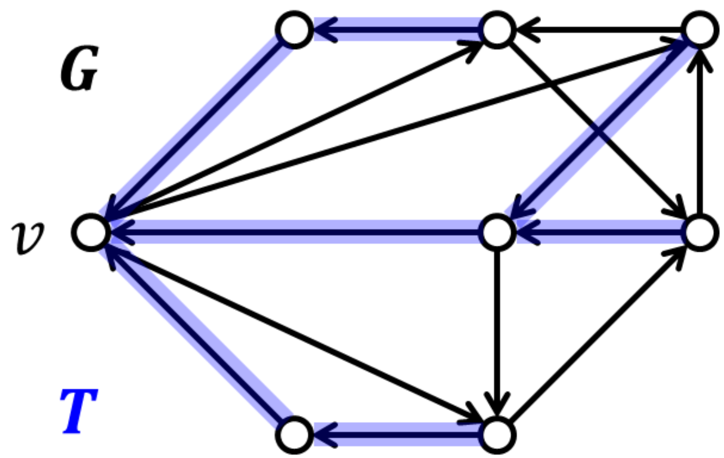
The algorithm starts traversal from v. We show that it must terminate at v and consume $E(G)$. Notice that all entering arcs of v belong to T.



The trail must terminate at v , since the traversal leaves a vertex $u \neq v$ along an edge $e \in T$ only after all the other out-arcs are consumed.

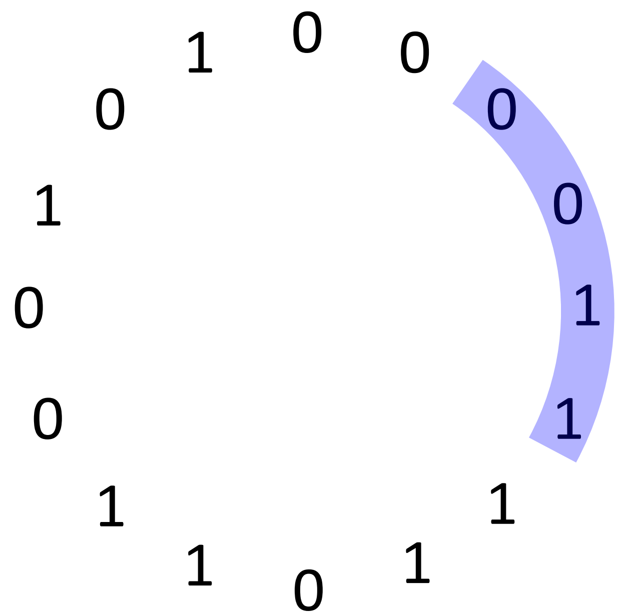
Since $d^+(u) = d^-(u)$, it implies that all u's in-arcs are also consumed, in particular, all those of T.

Therefore, entering into v with an arc e ($\in T$) implies that all the vertices and incident arcs of the sub tree rooted at v are consumed.



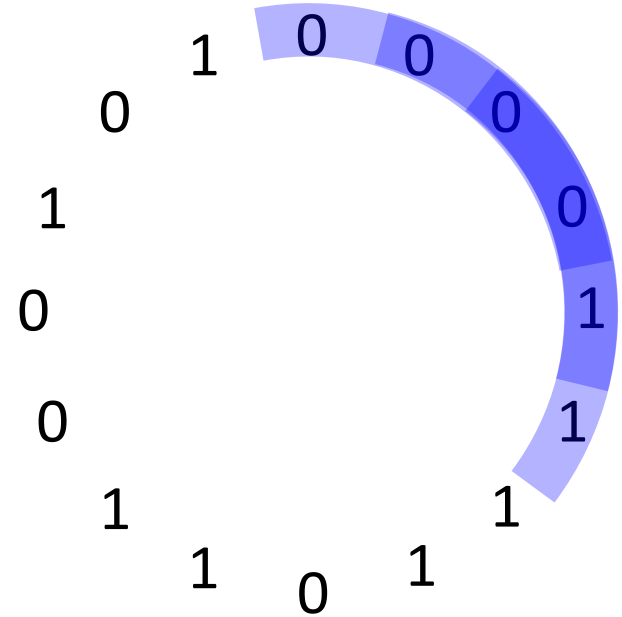
Finally, T spans G, so all the vertices, and hence arcs, must be traversed. ■

Application. Testing the position of a rotating drum.



Is there a cyclic arrangement of 2^n binary $0/1$ digits, such that the n-bit 2^n words obtained by successively sliding n-bit window are all distinct?

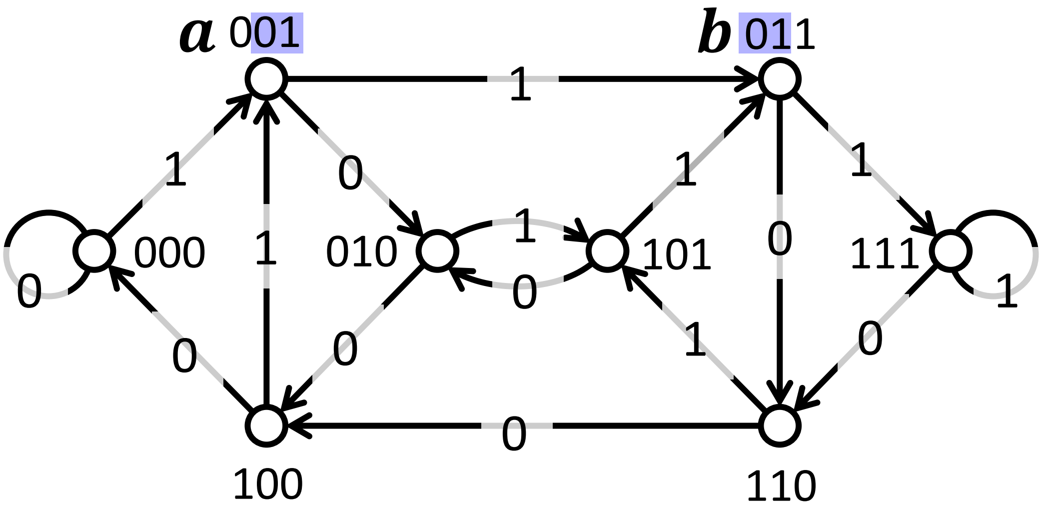
By encoding the currently read n-bit word (with a mounted sensor) the position of the drum is known.

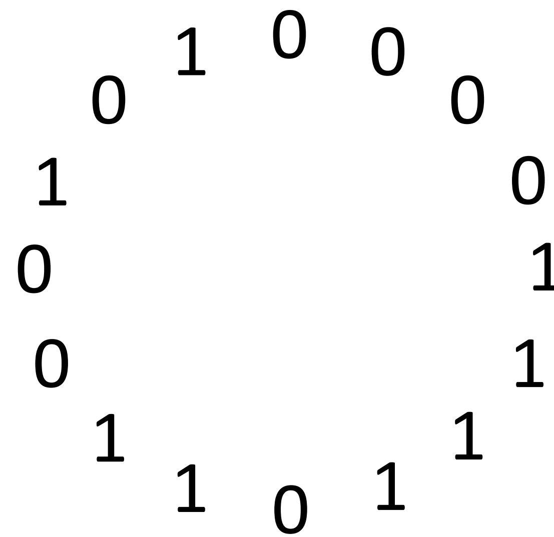


$\mathbf{0000111101100101}$ solves the problem for $n = 4$.

The problem can be solved using Eulerian digraph.

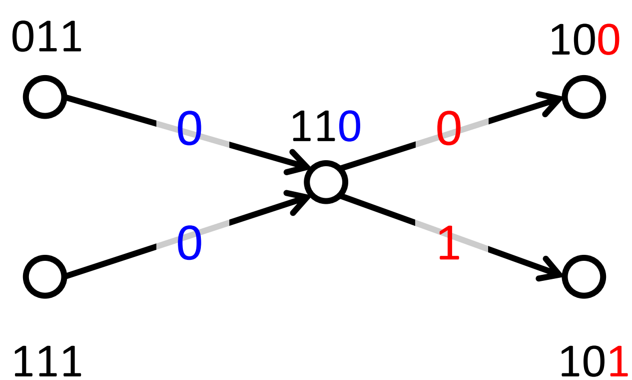
Associate the 2^{n-1} distinct $(n-1)$-bit words with the vertices of a digraph D_n (n-regular).

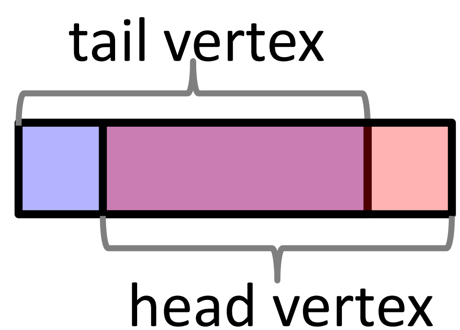




Place an arc from sequence $\boldsymbol{a}$ to sequence $\boldsymbol{b}$ if the $n-2$ **LSB**s of $\boldsymbol{a}$ agree with the $n-2$ **MSB**s of $\boldsymbol{b}$.

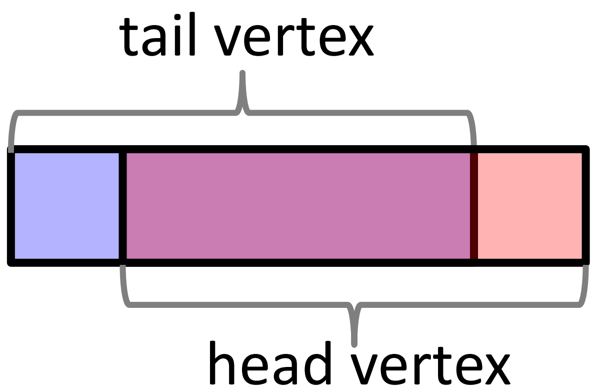
Label an arc with the LSB of b.





For each $(n-1)$-bit sequence (vertex) there are **two out-going** arcs labeled 0 and 1.

There are also **two in-coming** arcs labeled with the vertex's LSB. Hence D_n is **Eulerian**.



The n-bit codes obtained by appending the arc bit (LSB) to the $n-1$ bits at a tail vertex are all distinct. Two successive codes agree on the $n-1$ bits. ■

Application. Street-Sweeping Problem.

Curbs of a city are described by a digraph H.

Curbs are swept in the traffic direction. A **two-way** street implies two parallel **oppositely** directed arcs.

A **one-way** street implies two parallel arcs of **same** direction.

In NYC parking is prohibited from some street curbs each weekday to allow for street sweeping.

This defines a sub-graph G of H, consisting of the arcs available for sweeping on that day.

The problem is how to sweep G while **minimizing deadheading** time (no sweeping).

Each $e \in E(H)$ has a deadheading time $t(e)$.

If G is Eulerian no deadheading time is needed.

Otherwise, arcs of G are duplicated or arcs of H are added (not being swept).

Let $X \subset V(G)$ satisfy $d^{-}(x) > d^{+}(x)$, $\forall x \in X$, and $Y \subset V(G)$ satisfy $d^{+}(y) > d^{-}(y)$, $\forall y \in Y$.

Set $\alpha(x) = d^{-}(x) - d^{+}(x)$ and $\beta(y) = d^{+}(y) - d^{-}(y)$. There is $\sum_{x \in X} \alpha(x) = \sum_{y \in Y} \beta(y)$.

The Eulerian super-digraph must add $\alpha(x)$ arcs with tail in $x \in X$ and $\beta(y)$ arcs with head in $y \in Y$.

Since in the super-digraph the in and out degrees must be balanced, the additions should be paths in H from X to Y, which cost is the shortest path length.

This turns into a **Transportation Problem** with supplies $\alpha(x)$ for $x \in X$ and demands $\beta(y)$ for $y \in Y$.

The cost of shipment of one unit from x to y is $c(xy)$ (shortest path length), with $\sum_{x \in X} \alpha(x) = \sum_{y \in Y} \beta(y)$.

Transportation problem was introduced by Kantorovich (1939), solved by Hitchcock (1941) and many others. ■

Cuts and Connectivity

It is desired to preserve network service when some nodes or connections break.

For expensive connections it is desired to maintain connectivity preservation with as few edges as possible.

Graphs and digraphs are assumed loopless.

Definitions. A set $S \subseteq V(G)$ of a graph G is a **separating set** or **vertex cut** if $G - S$ has more than one component.

G is **$\boldsymbol{k}$-connected** if for every such S there is $|S| \geq k$.

The **connectivity $\boldsymbol{\kappa(G)}$** is the smallest vertex cut size.

Example. Though K_n has no separating set, we define $\kappa(K_n) = n - 1$.

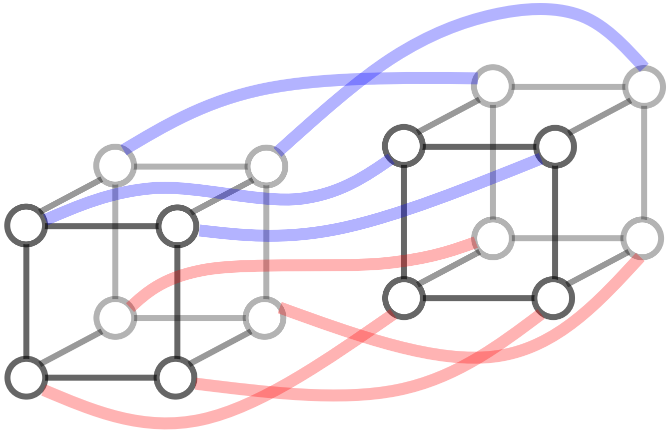
What is $\kappa(K_{m,n})$?

Every induced subgraph of $K_{m,n}$ having at least one $x \in X$ and $y \in Y$ is connected.

Hence either X or Y must be included in a separating set.

Since X and Y are separating sets by themselves, there is $\kappa(K_{m,n}) = \min\{m, n\}$. ■

Example. What is the connectivity of a k-dimensional cube Q_k?



A k-d cube has 2^k vertices, obtained from two Q_{k-1} copies by connecting matched vertices.

Q_k is k-regular. Deletion of the k neighbors of a vertex separates Q_k and hence $\kappa(Q_k) \le k$.

To prove that $\kappa(Q_k) = k$ we show by induction that every vertex cut has at least k vertices.

Q_k is obtained by matching the corresponding vertices of two Q_{k-1} copies Q and Q'.

Let S be any vertex cut of Q_k.

If $Q - S$ and $Q' - S$ are connected then $Q_k - S$ is connected too, unless S contains one end vertex of each of the 2^{k-1} matching edges.

But then $|S| \geq 2^{k-1} \geq k$ for $k \geq 2$, hence we may assume that $Q - S$ is disconnected.

Q has 2^{k-1} vertices, hence by induction $\kappa(Q) = k - 1$, hence S has $k - 1$ vertices in Q.

If S would not have vertices in Q' then $Q' - S$ would be connected and all the vertices of $Q - S$ have neighbors in $Q' - S$ (by the matching edges).

Q_k would therefore be connected unless S has at least one vertex in Q', yielding $|S| \geq k$. ■

When G is not a clique, deleting all the neighbors of a vertex disconnects G, so $\boldsymbol{\kappa(G) \leq \delta(G)}$, but equality does not necessarily hold.

Edge Connectivity

Perhaps that the transceivers (vertices) of a network G are so reliable that more than $\kappa(G) - 1$ never fail, hence communication is guaranteed.

It is desired that the links (edges) are also designed so it is hard to separate G by edge deletion.

Definitions. A **disconnecting set** of edges is a set $F \subseteq E(G)$ such that $G - F$ has more than one component.

Given $S, T \subseteq V(G)$, $[S, T]$ denotes the edges having one vertex in S and one in T. An **edge cut** is of the form $[S, \bar{S}]$, where $S \subset V(G)$.

G is ***k*-edge** connected if every disconnecting set has **at least** k edges.

The **edge-connectivity $\boldsymbol{\kappa'(G)}$** of G is the minimum size of a disconnecting set.

Every edge cut $[S, \bar{S}]$ is a disconnecting set since $G - [S, \bar{S}]$ has no path from S to $\bar{S}$.

The converse is false. The 3 edges of K_3 are disconnecting set, but not an edge cut. Still, there is:

Proposition. Every **minimal** disconnecting set is an edge cut.

Let $F \subseteq E(G)$ and $G - F$ have more than one component.

there must be some component H whose outgoing edges are deleted.

Hence F contains the edge cut $\left[V(H), \overline{V(H)}\right]$. F is not a minimal disconnecting set unless $F = \left[V(H), \overline{V(H)}\right]$. ■

Deleting one endpoint of each edge of F disconnects G.

It is therefore expected that $\kappa(G) \leq \kappa'(G)$ will always hold, unless a vertex deletion eliminates a component of $G - F$, producing a connected subgraph.

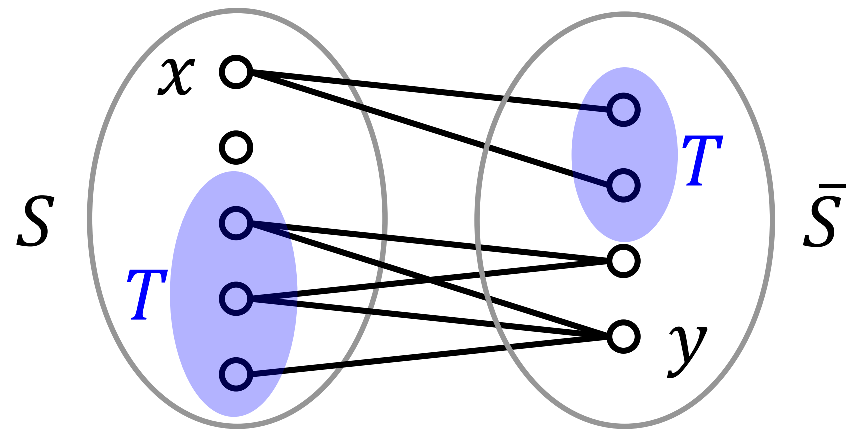
Theorem. $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof. $\kappa'(G) \leq \delta(G)$ follows immediately since the deletion of the incident edges of a vertex disconnects G.

To show $\kappa(G) \leq \kappa'(G)$ let $[S, \bar{S}]$ be a minimum edge cut.

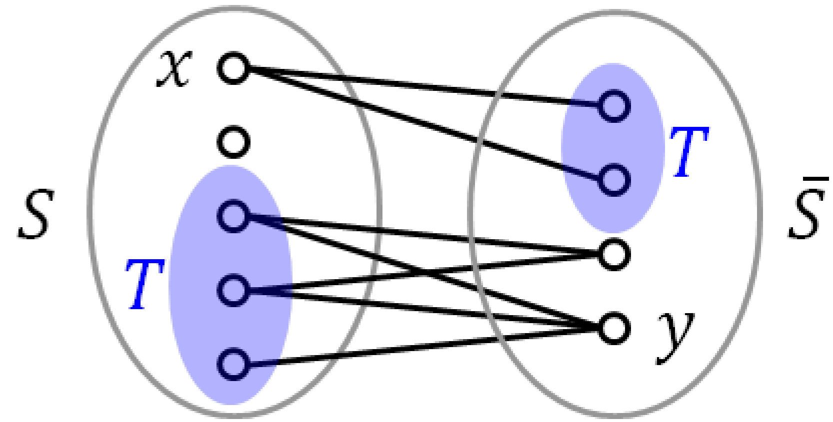
If every vertex of S is adjacent to every vertex of $\bar{S}$ then $\kappa'(G) = |S| \times |\bar{S}| \geq |V| - 1 \geq \kappa(G)$.

We therefore assume that there exist $x \in S$, $y \in \bar{S}$ but an edge xy does not exist.



Let T be the vertex set $(N(x) \cap \bar{S}) \cup (N(\bar{S}) \cap \{S - x\})$.

Since x and y belong to different components of $G - T$, T is a separating set.



The vertices of T can be associated with distinct edges connecting S to $\bar{S}$, hence $\kappa(G) \leq |T| \leq |[S, \bar{S}]| = \kappa'(G)$.

■

k-connected Graphs

A communication network is fault-tolerant if there are alternative paths between vertices.

The more vertex disjoint paths (except ends) the better.

Lemma. A graph G is connected iff for every non trivial partition $V(G) = S \cup T$, $S \cap T = \emptyset$, there is $st \in E(G)$ where $s \in S$ and $t \in T$.

Proof. Suppose G is connected, and let S and T be a partition.

Since G is connected, there is a path P between every $u \in S$ and $v \in T$.

Let $s \neq v$ be the last vertex of P in S and $t \neq u$ its successor. st is the desired edge.

Conversely, if G is disconnected, let H be a component of G. Then $S = V(H)$ and $T = V(G) - V(H)$ is a partition.

There cannot exist an edge between S and T, otherwise H would not be a component. ■

The above lemma shows that each pair of vertices is connected with a path iff G is 1-edge-connected.

We subsequently generalize this characterization to k-edge-connected graphs and to k-connected graphs.

Theorem. (Whitney 1932) A graph G, $|V(G) \geq 3|$, is 2-connected iff each $u, v \in V(G)$ are connected with a pair of internally-disjoint paths (disjoint vertices except u and v).

Proof. Suppose that any two vertices are connected with a pair of internally-disjoint paths.

Deletion of one vertex cannot disconnect these vertices, and at least two vertex deletion is required, hence *G* is 2-connected.

Conversely, suppose that G is 2-connected. We prove by induction on $d(u, v)$ (shortest path) that two internally-disjoint (u, v)-paths exist.

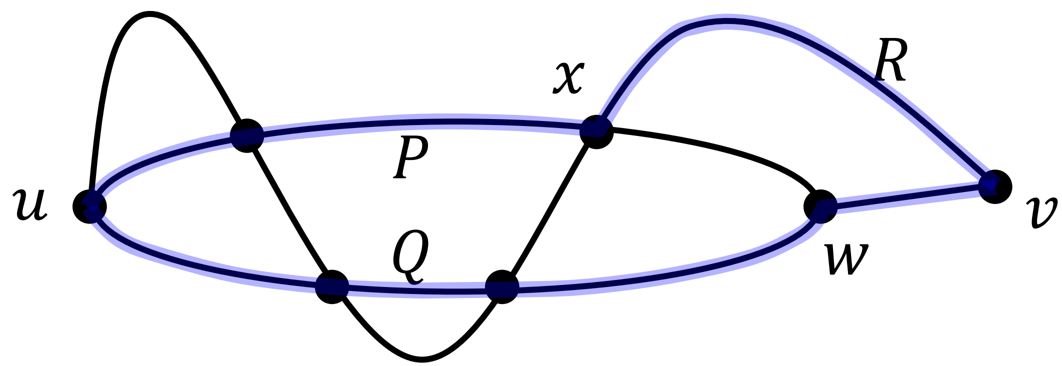
For the base $d(u, v) = 1$. $G - uv$ is still connected, since $\kappa'(G) \geq \kappa(G) \geq 2$ (G is 2-connected).

A uv-path is internally-disjoint from uv, which being an edge, has no internal vertices, hence two disjoint paths exist.

Assume by induction that G has a pair of internally-disjoint xy-paths for $r > d(x, y) \geq 1$.

Let $d(u, v) = r$ and let w be the vertex before v on the uv-path.

$d(u, w) = r - 1$, and by induction there are uw vertex-disjoint paths P and Q.



Since $G - w$ is connected, there is a uv-paths R, $R \neq P$ and $R \neq Q$.

If R is vertex-disjoint of P or Q we are done, since R and either of $P \cup wv$ or $Q \cup wv$ are edge disjoint paths. .

Otherwise, let R intersect both P and Q.

Assume w.l.o.g that its last common vertex x is on P.

Combining the ux-path of P with xv-path of R yields a vertex-disjoint path to $Q \cup wv$ path. ■

Theorem. (2-connected graph characterization)
If $|V(G)| > 3$, the following conditions are equivalent.

- G is connected and has no cut-vertex.
- For all $x, y \in V(G)$, there are internally-disjoint xy-paths.
- For all $x, y \in V(G)$, there is a cycle through x and y.
- $\delta(G) \geq 1$, and every pair of edges of $E(G)$ is on a common cycle.

Let $\boldsymbol{G(x,y)}$ be a graph with two distinguished vertices.

The **local connectivity** $\boldsymbol{p(x,y)}$ is the maximum number of pairwise internally disjoint xy–paths. $\boldsymbol{c(x,y)}$ denotes the **smallest vertex cut** separating x and y.

Theorem. (Menger 1927, Göring's proof 2000)
Let $x, y \in V(G)$ be non adjacent. Then $p(x, y) = c(x, y)$.

Proof. Let $c_G(x, y) \coloneqq k$. Let us show that $p_G(x, y) \leq k$.

Let Π be a set of internally disjoint xy–paths . Each path of Π must meet at least one vertex of any xy-vertex-cut, as otherwise x, y would have been connected.

Hence, the xy-vertex-cut must have at least $|\Pi|$ distinct vertices, yielding $p_G(x, y) \leq k$.

We subsequently show by induction on $|E(G)|$ that $p_G(x,y) \geq k$.

We can assume that there is an edge $e = uv$ incident neither to x nor to y.

Otherwise, every xy–path is of lengths 2, so the interim vertices of the xy-paths define an xy-vertex-cut and the Theorem's conclusion follows immediately.

Set $H \coloneqq G \backslash e$. Because H is a subgraph of G there is $p_G(x,y) \geq p_H(x,y)$.

By induction there is $p_H(x,y) = c_H(x,y)$.

Since every xy-vertex-cut of H together with either end of $e = uv$ is an xy-vertex-cut of G , there is $c_G(x, y) \leq c_H(x, y) + 1$, yielding

$$p_G(x, y) \geq p_H(x, y) = c_H(x, y) \geq c_G(x, y) - 1 = k - 1.$$

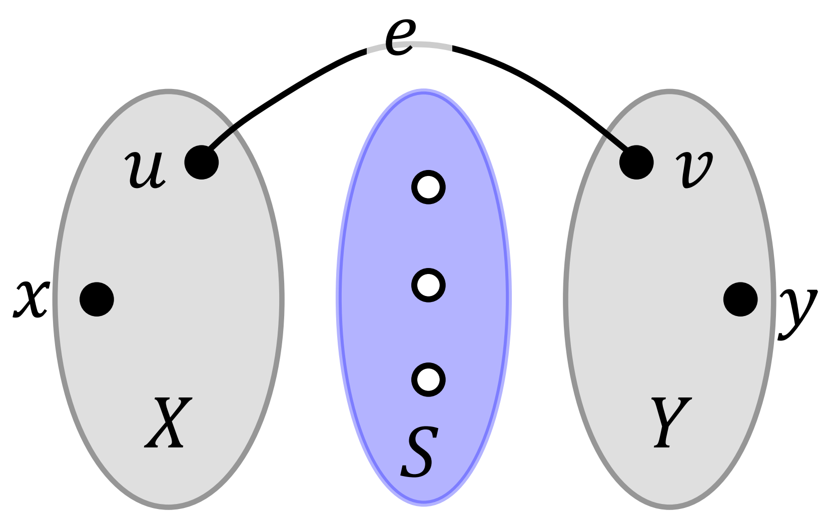
We may assume that $p_G(x, y) = k - 1$.

Otherwise, $p_G(x, y) > k - 1$, hence $p_G(x, y) \geq k = c_G(x, y)$ and the Theorem's conclusion follows.

Thus, in particular, $c_H(x, y) = k - 1$.

Let $S \coloneqq \{v_1, \ldots, v_{k-1}\}$ be a minimum xy-vertex-cut in H.

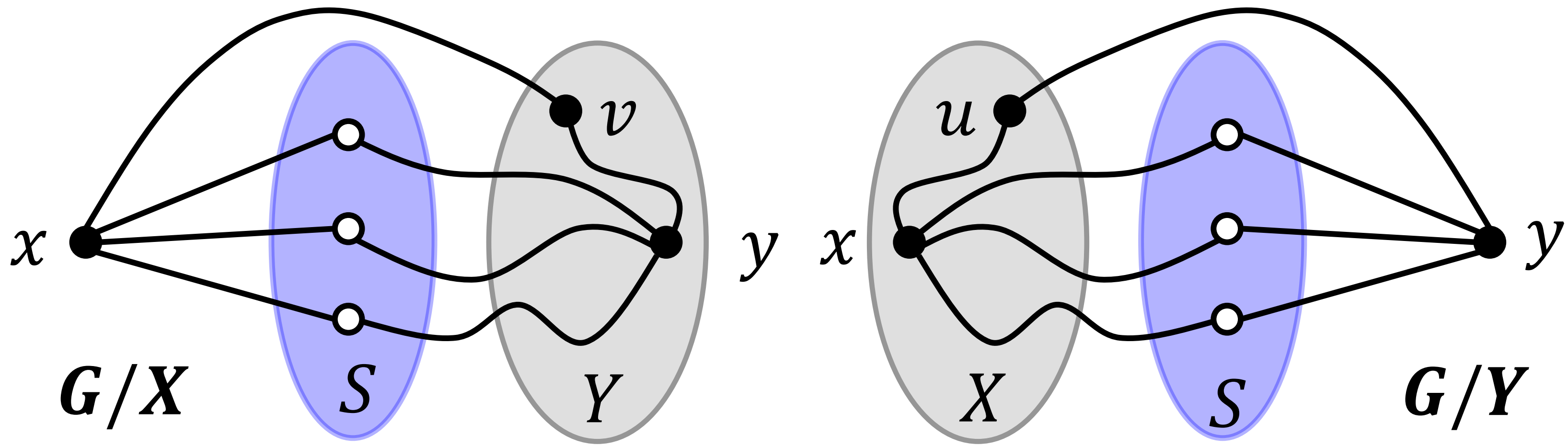
Let $X(Y)$ the set of vertices reachable from $x(y)$ in $H - S$. There is $X \cap Y = \emptyset$.



Since $|S| = k - 1$, S cannot be an xy-vertex-cut of G ($c_G(x, y) \coloneqq k$).

Therefore, there must exist an xy-path in $G - S$, that includes $e = uv$, where w.l.o.g $u \in X$ and $v \in Y$.

Let us contract Y into a single vertex y, and denote the outcome by G/Y.



Likewise, let us contract X into a single vertex x, and denote the outcome by G/X.

Every xy-vertex-cut T in G/Y is necessarily so in G. Otherwise, there was an xy-path P in G avoiding T.

The subgraph P/Y of G/Y would then contain an xy-path in G/Y avoiding T, impossible since T is an xy-vertex-cut in G/Y.

Consequently $c_{G/Y}(x, y) \geq k$ ($c_G(x, y) \coloneqq k$).

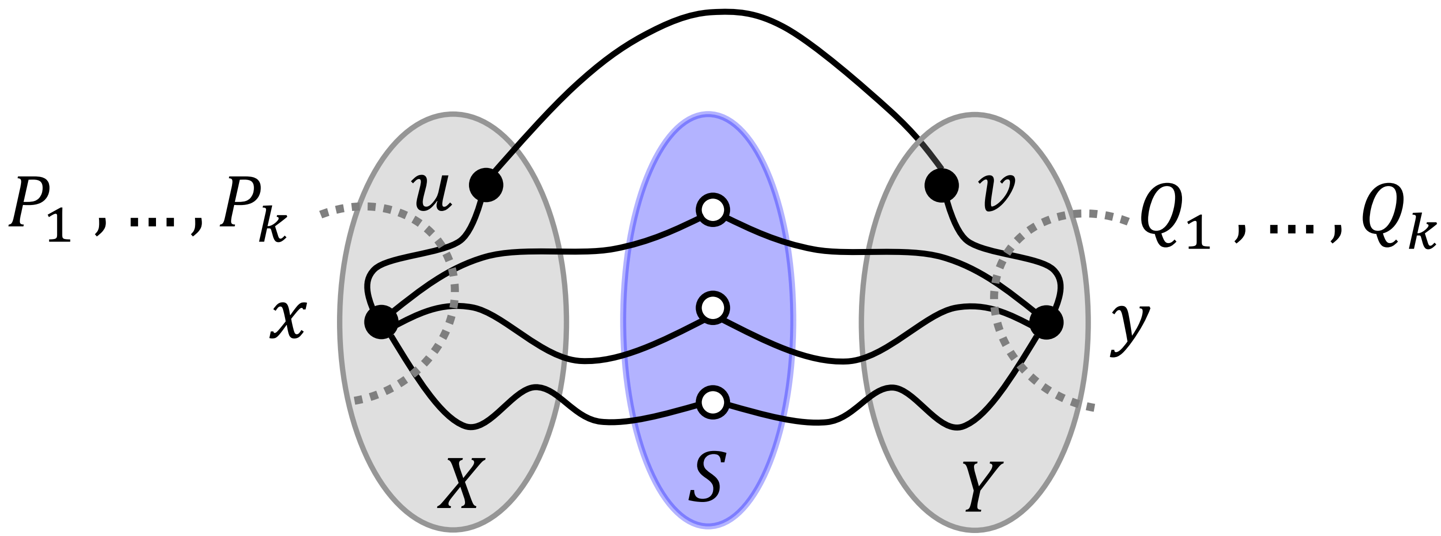
On the other hand, $S \cup \{u\}$ which is an xy-vertex-cut of G, is also such in G/Y, and therefore $c_{G/Y}(x, y) \leq k$.

Consequently, $c_{G/Y}(x, y) = k = |S \cup \{u\}|$, and $S \cup \{u\}$ is a minimum xy-vertex-cut in G/Y.

By the induction hypothesis ($|E(G/Y)| <$ ($|E(G)|$), there are k internally disjoint *xy*-paths $P_1, \ldots, P_k$ in G/Y, and each $v \in S \cup \{u\}$ must lie on one and only of them.

Assume w.l.o.g that $v_i \in V(P_i), 1 \leq i \leq k-1$, and $u \in V(P_k)$.

Likewise, there are k internally disjoint xy-paths $Q_1, \ldots, Q_k$ in G/X, obtained by contracting X to x, such that $v_i \in V(Q_i), 1 \leq i \leq k-1$, and $v \in V(Q_k)$.



It follows that there are k internally disjoint xy-paths in G, $xP_iv_iQ_iy, 1 \leq i \leq k-1$, and xP_kuvQ_ky.

Consequently $p_G(x,y) = k$. ∎

Example. Show that $c(G) + |E(G)| \geq V(G)$ for every G, where $c(G)$ is the number of connected components.

Proof. By induction on $|E(G)|$. If $|E(G)| = 0$ G has only isolated vertices, so $c(G) = V(G)$ and an equality holds.

Let $e \in E(G)$. Since the removal of an edge can turn a connected component into two, there is

$$(1) \quad c(G) \geq c(G - e) - 1.$$

Assume by induction that

$$(2) \quad c(G - e) + |E(G - e)| \geq V(G).$$

Substitution of (2) in (1) yields

$$c(G) \geq c(G-e) - 1 \geq$$
$$V(G) - |E(G-e)| - 1 =$$
$$V(G) - (|E(G)| - 1) - 1 = V(G) - |E(G)|. \blacksquare$$