The Marriage Theorem

2.1 Given a set of girls, a set of boys, and the set of pairs, (i, j), such that boy, i, loves girl, j. (All explicitly mentioned sets in these lectures are finite unless explicitly clear otherwise.)

The traditional way to say the marriage theorem is:
The girls can all marry distinct boys who love them iff
for, every subset S of the girls, the size of S is not smaller than the number of boys who love someone in S.

I prefer the following way to say the theorem because to me it is weird to speak of **all** subsets of girls as a criterion for the conclusion.

For any input to the theorem, either there is a way for all the girls to marry distinct boys who love them, or else there is a subset S of the girls which is smaller than the number of boys who love someone in S. (Not both).

We will see that the best way to prove the non-obvious part of the marriage theorem is by an algorithm which tells how to find either one way for the girls to all marry,
or else one subset S of the girls.

The non-obvious part of the marriage theorem is an example of an "existentially polytime theorem" (EP theorem).

The "not both" part of the marriage theorem is not EP.
An EP theorem for which we also have "not both"
is called "a good characterization". A good characterization leads to what is called an "NP-coNP" predicate.

Roughly speaking, an EP theorem is one which says that, for any input, there exists something, any instance of which, when appropriately displayed, can easily be seen to be an instance.

(2.1.2) Not all EP theorems have "not both".
For example a theorem of Dirac says:
For any graph G, either there is a node v in G whose "degree"
(the number of edges of G which meet v)
is smaller than half the number of nodes of G,
or else there is a hamilton cycle C in G
(i.e., a way to tour all the nodes **once** and return home by
walking along edges of G).

Maybe there exists both a node v and a cycle C.
An algorithmic proof needs only to tell how to find one instance of
one or the other. It might find a hamilton path when there is a
node of "degree" smaller than half the nodes. Or vice versa it
might find a node of small degree when there is a hamilton
polygon.

2.2   We will give five very different proofs of the marriage theorem. The first is transparent and algorithmically useful. The other four proofs are algorithmically, in themselves, not very useful.

One reason we give them is to illustrate that proofs which are traditionally appealing can leave the impression that there is no easy way to find an instance of what an EP theorem says exists. It is curious that mathematicians traditionally have liked to prove things exist but have had less interest in how to find them.
All five of the proofs are instructive in five very different ways.

Before giving the proofs, we mention **some mathematically popular ways to represent various theories including the marriage theorem**. They sometimes even cause mathematicians to imagine that they are treating different theories, though they are not really different theories in the sense that it is easy to translate from one representation to another.

The 'bipartite graph' representation, K, has a 'node' for each girl, a 'node' for each boy, and an 'edge' for each girl-boy pair where the boy loves the girl.
The 'subset collection' representation has the set, B, of boys and, for each girl, the subset of boys who love her.
The 'incidence matrix' representation has a matrix whose columns are the girls, whose rows are the boys, and whose entry in each position (i,j) is non-zero when boy i loves girl j, and zero otherwise.

2.3 Here is the proof which describes an easy algorithm

to either find a way for all the girls to have distinct husbands who love them or else find a find subset of girls who together are not loved by enough boys.

Let's do it in terms of the graph representation.
A 'matching' in the graph K is a subset M of the edges such that any node of G (boy or girl) is 'hit' by at most one member of M. We want to see how to find either a matching which hits all the girl nodes or else a subset of the girl nodes which is bigger than its neighboring set in K of boys.

Suppose we have a matching M which does not meet girl, g. Grow an 'alternating tree' T in K, two edges and two nodes at a time, as follows, starting with node g (which by itself is an alternating tree): choose to add to T, if possible, any edge e which hits a girl in T and a boy, say b, not in T, and then add to T, if there is one, the edge of M which hits node b.
If you find that there is no edge e, then the set of girl nodes in T has a too small neighboring set in K of boys.

If there is an edge e, but the other end b of e is not hit by an edge of M, then trace the path P in [T + e] from b back to g. Get a larger matching M' by replacing the edges of M in P by the other edges of P. That's it, the proof and easy algorithm.
In the live performance, we will show pictures.
But here, you should draw pictures.

2.4 The Halmos-Vaughn proof of the marriage theorem:

Given K, assume the marriage theorem has been shown to be true for any bipartite graph K' which has fewer girl nodes than K. Suppose there is no subset S of the girls of K whose neighboring set of boys in K is smaller. We want to see that there is a matching M in K which hits all the girls of K.

If the neighboring set of every proper subset S of the girls of K is larger than S, we can choose any edge e hitting any girl for our matching, and then find the rest of M as any matching which hits all the girls in the graph K', obtained from K by deleting e and the boy and girl hit by e.

Otherwise, there is some proper subset S of girls in K whose neighboring set of boys in K is the same size as S. Take any matching M' in K which hits all of S. In the graph K', obtained from K by deleting M' and all the nodes hit by M', there is a matching which hits all the girls of K'. It together with M' is a matching in K which hits all the girls of K. That's it.

2.5 A linear algebra proof of the marriage theorem:
Consider the matrix K whose columns are the girls, whose rows are the boys, and whose non-zeroes are all different letters, i.e., algebraically independent 'indeterminates', in the positions (i, j) such that boy i loves girl j.

Since the determinant of a square matrix means the product of entries in a general diagonal summed (with +1 or -1 coefficients) over all general diagonals, a subdeterminant of K is different from zero iff it has a general diagonal all of whose entries are non-zero. Hence the girls can all marry distinct boys who love them iff some square submatrix of K, using all the columns, has non-zero determinant.

If this is not the case then, by linear algebra, there is some minimal subset say S of columns which is linearly dependent. That is, the coefficients are non-zero in a linear combination of S equal to the column of all zeroes.

By familiar linear algebra, these coefficients are subdeterminants of a matrix whose columns are all the columns of S and whose rows are some subset, say R, of the boys which is smaller than S.

If the columns of S had non-zeroes in some row not in R, then since we have a linear combination, whose coefficients are non-zero subdeterminants of K, which equals zero, we would have a non-trivial algebraic relation of the algebraically indeterminate letters in K. Hence there is no boy not in R who loves a girl in S, and so the number of boys who love some girl in S is smaller than S. That's it.

One could use a version of Gaussian elimination to calculate the rank of K, and thus determine the largest number of girls who can marry. **However** the algorithm would generate polynomials in the indeterminates whose sizes grow exponentially with the input, K, and so it's a very bad algorithm.

On the other hand, the algorithm in our first proof of the marriage theorem can be used to compute in polytime the linear algebra rank of K, that is, the largest number of girls who can marry. (We will see how to do it in the next section on the Hungarian Method.)

In a later chapter we will study Gaussian elimination in order to learn a version which is polytime for matrices of rational numbers.

A 'matroid', (E,F), is a groundset E, and a non-empty family F consisting of the so-called 'independent subsets' of E, such that every subset of an independent set is independent,
and, for every subset S of E, every inclusion-maximal subset of S **is the same size, called the rank r(S) of S, in matroid (E,F).**

If E is the column set of a matrix whose entries are numbers, or polynomials in algebraic indeterminates, and if F is the family of linearly independent subsets of E, then (E,F) is a matroid.

(2.6.1) Our linear algebra proof of the marriage theorem observes, and uses, that for any instance with set E of girls, and with members of F being the subsets of E which have a set of distinct boys who love them, (E,F) is a matroid. These are called 'transversal matroids'.

2.7 A linear programming proof of the marriage theorem:

A matrix is called 'totally unimodular' (t.u.) when every subdeterminant equals 0, 1, or -1.
If a matrix A is such that, for each column, the entries in that column are zeroes except for at most one -1 and at most one 1, then it is fairly obvious that A is t.u. It also obvious that a matrix obtained by adjoining an identity matrix to a t.u. matrix is still t.u.

A 'digraph', G, is a set, V, of 'nodes' and a set, E, of 'edges', such that each edge has a single 'tail' in V and a single 'head' in V. The **incidence matrix**, A^1, of G has a column for each edge of G and a row for each node of G. The entry in position (i,j) is +1 when node i is the head of edge j, -1 when node i is the tail of edge j, and 0 otherwise. Hence A^1 is t.u.

Where A^1 and A^2 are any matrices of real numbers,
where b^1 and b^2 are compatible column-vectors of real numbers,
and c is a row-vector of real numbers, the general
linear programming problem can be formulated as:

(2.7.1) Find, if there is one, a column vector x such that
$A^1 x = b^1$; $A^2 x \leq b^2$; and such that cx is maximum (or minimum).
(Usually the inequalities $x \geq 0$ are included.)

A point x is called a **basic point** of the linear system,
$L = \{A^1 x = b^1, A^2 x \leq b^2\}$,
if it is the unique solution of the set of constraints of L
which it satisfies with equality.
A point x is called a feasible point of L if it satisfies L.
$P(L) = \{x: A^1 x = b^1, A^2 x \leq b^2 \}$ is called a (convex) **polyhedron**.
It is called a **polytope** when it is bounded.

(2.7.2) Some basic facts of linear programming:

(2.7.2a)   A point x^0 is a basic feasible point of a linear system L iff there exists a linear objective, cx, such that x^0 is the unique point that maximizes cx over P(L), the solution set of L. Hence, the b.f. points of L are also called the **extreme points** of P(L), or the vertices of P(L)
.
(2.7.2b)   Every linear function, cx, of x in P(L) is maximized over P(L) by a basic point of L and so, in principle, a finite algorithm for the linear programming problem is simply to check each basic point for its feasibility, and how big it makes cx.

(2.7.2c) The l.p. duality theorem:
The **'dual l.p.'** of l.p. **(2.7.1)** is the l.p.:
Find, if there is one, a rvector $y = (y^1, y^2)$ such that
$y^2 \geq 0$; $y^1 A^1 + y^2 A^2 \geq c$; and $y^1 b^1 + y^2 b^2$ is minimum.
If **(2.7.1)** and it's dual have feasible solutions, then they both have basic feasible solutions which achieve max cx = min yb.

Where A^1 is the incidence matrix of a digraph, and A^2 has only one non-zero, ±1, in each column,
the l.p. (2.7.1) is called
a **'network flow problem'**, or a **'circulation problem'**.

By total unimodularity, and Cramer's rule for solving a system of linear equations, if b is integer valued
then the basic points of l.p. **(2.7.1)** are integer valued,
and if c is integer valued then the basic points of the dual l.p. are integer valued.
Hence, if both b and c are integer valued, then **(2.7.1)** is solved by an integer valued **flow**, x, and its dual is solved by an integer valued **dual-flow**, $y = (y^1, y^2)$.

By appropriately specializing, we get the marriage theorem. (Homework). That's a lot of machinery just to prove the marriage theorem. However, operations researchers are justified in taking this proof very seriously, because at the same time it proves a great many other combinatorial theorems.
There are thousands of papers on solving network flow problems. We'll study the proto-typical instance in the section here on the Hungarian method.

2.8 A **submodular function**, f, is real-valued function of the subsets of a set E which satisfies f(A∩B) + f(AUB) ≤ f(A) + f(B) for all subsets A and B of E.
A submodular function f is called a **polymatroid function** if we also have 0 = f(**ø)** ≤ f(A) ≤ f(AUB) for all subsets A and B of E.

(2.8.1) It is easy to show that the rank function of some matroid on ground set E is the same thing as an integer-valued polymatroid function, f, where f of each singleton is 1 or 0. For any such f, the independent sets of the matroid, whose rank function is f, are the subsets J of E whose cardinality is f(J).

The following theorem provides **a way to derive matroids which has nothing to do with algebraic independence**.
(2.8.1) Theorem. For any polymatroid function of subsets of E, let J be an independent set (i.e., J ε F), when |S| ≤ f(S) for every subset S of J.
Then (E, F) is a matroid.

Assuming we have an oracle which for any S tells us the value of f(S), we can demonstrate that a set J is not independent in the matroid (E, F) of theorem (2.8.1) by showing one subset S of J which is bigger than f(S). How can we demonstrate that a set J **is** independent without calling the oracle for every subset S of J? In a later lecture we will study this question generally, but here we answer the question for an important special submodular function:

(2.8.2) Theorem. Where K is a bipartite graph with the set B of boy nodes, the set E of girl nodes, and an edge from boy to girl when the boy loves the girl, let f(S), for any subset S of the girls, be the number of boys who loves one or more of the girls in S. Then f is polymatroid function of the subsets of girls. (Homework.)

The marriage theorem says that we can always demonstrate that a subset J of the girls is independent in the matroid determined by the polymatroid function of Thm (2.8.2) by showing a way that the girls of J can marry distinct boys who love them.

A submodular proof of the the marriage theorem: Where f(S) is the polymatroid function as in (2.82), suppose we have that |S| ≤ f(S) for every subset S of J. If each girl in J is loved by exactly one boy then clearly the girls are loved by distinct boys, and the marriage theorem is proved.
If there are two boys who love the same girl, we can use submodularity to show that one of those boys can stop loving the girl and we will still have |S|≤ f(S) for every subset S of J. (Homework.) That's it.

This matroid (E, F) of Thm (2.8.1), where f is as in Thm (2.8.2), is the same 'transversal matroid' as defined in (2.6.1). In fact the desire for an application of Thm (2.8.1), rather than (2.6.1), is how transversal matroids were discovered.