Spectral Graph Theory | Instructor: Padraic Bartlett

Lecture 5: Applications! (I.e.: The Google, How Does It Work.)

Week 4 | *Mathcamp 2011*

Last lecture! In our first week of spectral graph theory, we focused on studying just what the spectra told us about a graph, and found a series of spectral answers to the kinds of questions (what can we bound the chromatic number by, how does the spectrum relate to the degree, how can we tell whether a graph is bipartite) we typically ask in an introduction to graph theory course. From there, we moved to using spectral graph theory as a tool to answer graph-theoretic questions in their own right: in the span of a few days, we came up with a beautiful formula for the number of spanning trees in any graph, and used the Lagrangian to show that many graph decompositions are impossible (K_{10} into three Petersen graphs, K_n into $n-2$ bipartite graphs) and some graph structures are inevitable (K_r is inescapable if you have $\geq \frac{r-2}{r-1}n^2$-many edges.) From there, we moved to more impressive results: via the integrality conditions on strongly regular graphs, we managed to prove the Friendship Theorem, and show that there are at most five graphs that are "like" the Petersen graph (with the existence of the fifth still an open question!)

Today, I want to sketch briefly how non-mathematicians use the tools in spectral graph theory to attack "real-world" problems. Graphs come up everywhere, and analyzing them is hardly a task reserved only for graph theorists; in this lecture, we'll examine how spectral graph theory can tell us which fullerenes seem to be theoretically constructible and which aren't, and how the Google Page Rank algorithm works.

1 Fullerenes!

In graph theory, a **fullerene**[1] is a 3-regular planar graph in which all of its faces are 5- or 6-cycles, including the "outer" face[2]. Observant or chemically-trained readers will recognize this name from chemistry, where a fullerene is any carbon molecule that forms a sphere or ellipsoid; the connection here arises from the observation that spheres of carbon can pretty much only be made when the carbons are joined in cycles of 5 or 6.

A question that both graph theorists and chemists are fairly interested in is the following: what kinds of properties do fullerenes have? What kinds of graphs can be fullerenes? Which graph-theoretic fullerenes are also viable molecules in reality?

Surprisingly, we can answer a lot of these questions! We'll defer a number of proofs, as they involve some linear algebra we haven't had the chance to develop yet; but the feel for the material should hopefully be conveyed. Unless explicitly stated, a fullerene will be a graph-theoretic fullerene for the purposes of our discussion.

Proposition 1 *Any fullerene has precisely 12 5-cycles.*

[1] In popular science fiction, a **fullerene** is a plot device that basically allows you to do whatever you want. See also: wormholes, quantum mechanics, hot cups of tea.

[2] In this sense, we are considering our planar graph as a graph we can embed on the sphere without crossings, in which every face is a C_5 or C_6.

Proof. Take any fullerene with V vertices, E edges, and F faces, and recall what the Euler characteristic says about any graph embedded on a sphere:

$$V - E + F = 2.$$

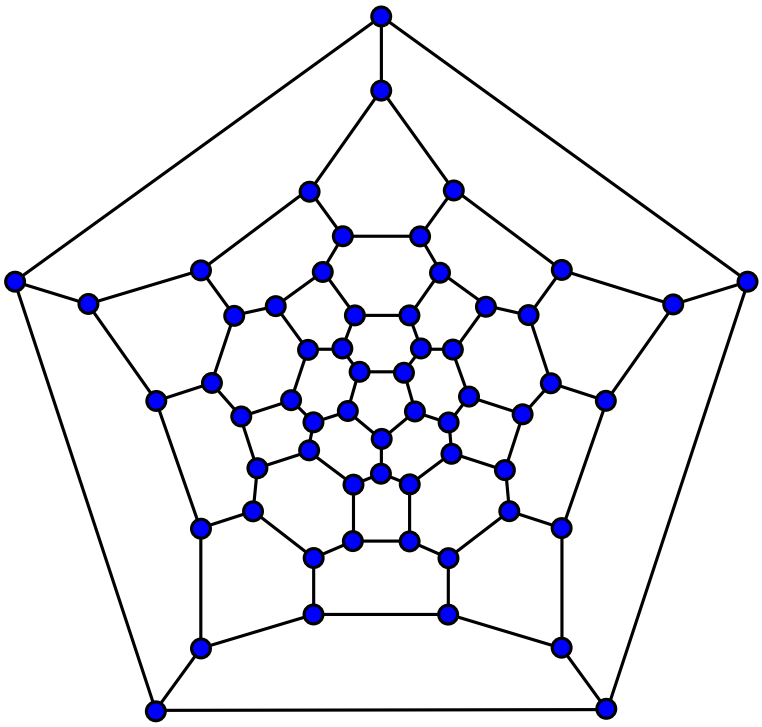
We know that our graph is 3-regular; therefore, we have $3V = 2E$ via the degree-sum formula. Combining this with the Euler characteristic, we have that $F_5 + F_6 = F = 2 - V + E = 2 + \frac{1}{2}V$ Furthermore, because every edge lies in precisely two faces, we have that $5F_5 + 6F_6 = 2E = 3V$. Subtracting 6 copies of the first equation from the second yields $-F_5 = -12$, i.e. $F_5 = 12$, our claim.

Notice that as a quick consequence we can relate the number of 6-cycles and vertices via $n = 2f_6 + 20$, as well.

In chemistry, it seems that not every fullerene is realizable. In particular, one rule that chemists have noticed that all fullerenes obey is that **they never have two adjacent pentagonal faces**: this is probably because the pentagon is not a shape that carbons are terribly happy in, and the stress of having any carbon in two such faces probably makes any such molecule unstable.

Therefore, it seems likely that any viable fullerene will have to have all of its pentagonal faces isolated. By the proposition above, it must have at least 60 vertices, as it has precisely 12 such faces. Does such a fullerene exist?

The face-meltingly awesome answer lies below:



As an added bonus, this one **actually exists** in reality! It's called **buckminsterfullerene**, after Richard Buckminster Fuller, who is an awesome person you should read about on Wikipedia.

Heartened by our success, we might ask the following: can we make more? Specifically, if we find other such fullerenes that satisfy this property, will they too exist in nature? Frustratingly, this seems to not be true: there are many fullerenes that do not have adjacent pentagons that still don't seem to exist.

One necessary condition chemists have noticed about fullerenes that seem to exist (and a lot of molecules in general) is that their adjacency matrices seem to need to have as many positive eigenvalues as they have negative eigenvalues: this seems to be the consequence of some sort of need for a "balancing" condition, where having too many eigenvectors in either direction makes the molecule unstable. This winds up disqualifying tons of potential other fullerenes.

At this point, we might despair; as we've seen in this class thus far, finding eigenvalues is a tricky bit of business to do, and doing so on an ad-hoc basis seems difficult, especially for graphs that are all on ≥ 60 vertices! However, consider the following beautiful process:

Definition. Given a 3-regular planar graph G with v vertices, e edges, and f faces we define the **leapfrog graph** $F(G)$ of G as follows:

1. Start with G.
2. Turn G into its line graph $L(G)$, which is formed by taking G's edge set as its vertices and connecting two edges iff they are incident in G. This turns G into a 4-regular planar graph, as can be seen in the example drawn below, with m vertices and $n + f$ edges.
3. Now, take this graph and split each vertex of $L(G)$ into a pair of adjacent vertices in such a way that every triangular face of $L(G)$ is now a hexagon. This then gives us a 3-regular planar graph with $2m$ vertices, n faces of length 6, and the same number of non-hexagonal faces as the original graph: i.e. if we put in a fullerene, we'll get out a fullerene!

We don't quite have the linear algebraic tools to prove it here, but (as it turns out) these leapfrog graphs *always* turn out to have this stability property (that they have as many positive eigenvalues as negative eigenvalues!) We state the proposition here; find me in TAU if you want more information about the techniques needed[3] to prove this result:

Proposition 2 *If X is a fullerene graph, then $F(X)$ is a a fullerene with as many negative eigenvalues as positive eigenvalues.*

2 The Google

Perhaps one of the most well-known applications of spectral graph theory is Google's PageRank algorithm. Basically, before Google came along, web search engines were **atrocious**[4];

[3] Particularly interested students are invited to first learn about the process of **interlacing**, which is described nicely in Brouwer/Hamer's online notes on spectral graph theory, and then to find Godsil/Royle's book on algebraic graph theory, where this is carefully proved.

[4] Please, please tell me that some of you remember a world before Google was known? Really? AAAAAAAA I AM OLD

search results were basically massive keyword-bashes plus some well-meaning but dumb attempts to improve these results by hand. Then Brin and Page came onto the scene, with the following simple idea:

Important websites are the websites other important websites link to.

This seems awfully circular, so let's try framing this in more of a graph-theoretic framework: Take the internet. Think of it as a collection of edges[5] connecting n webpages together: the various webpages are vertices, and there are edges pointing out of each website to each page that site links to. In this sense, if we have some quantity of "importance" $rank(v_i)$ that we're associating to each webpage i, we want it to obey the following relation:

$$rank(v_i) = \sum_{v_j \in N^-(v_i)} \frac{rank(v_j)}{\deg(v_j)}.$$

Hmm. Still circular. But more promising! In particular, if we don't think of each ranking individually, but rather take them all together as some large rank vector $\mathbf{r}$, we have that $r_k = \sum_{i=1}^{n} a_{ji} \frac{r_i}{\deg(v_i)}$: in other words,

$$\mathbf{r} = A_G^T \cdot D_{\deg^+}^{-1} \mathbf{r},$$

where $D_{\deg^+}^{-1}$ has the reciprocals of the out-degrees of the vertices on its diagonal. But what does this mean that $\mathbf{r}$ is? Just $\mathbf{r}$ an eigenvector for the eigenvalue 1 of the matrix $A_G^T \cdot D_{\deg}^{-1}$!

Ok, that's slick. There are some issues with this, however: in the event that any vertex has 0-out degree, the inverse of the diagonal matrix will (um) not exist. Also, we would kind of like our ranking to be a well-defined thing: if there were twelve distinct notions of ranking, for example, we often might not be able to tell which ranking was more relevant to us at any point in time. For general matrices, this isn't the case: if we look at $K_2 \coprod K_2$, the disjoint union of 2 K_2's, we get the adjacency matrix

$$A_G = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\Rightarrow A_G \cdot D_{\deg^+}^{-1} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

which has both $(1, 1, 0, 0)$ and $(0, 0, 1, 1)$ as eigenvectors for 1.

So: how can we fix this? Well: one idea is to rethink the above process slightly. What we're trying to model here is an idea of importance for websites: i.e. if we're on the internet and wandering through webpages, we should in theory be clicking on links with the

[5] A "series of tubes," if you will.

probabilities given to us by the $rank(v_i)/\deg(v_i)$'s. In other words, the matrix $A_G^T \cdot D_{\deg^+}^{-1}$ is a **probability matrix**: if we multiply it on the right by any vector whose coördinates sum to 1 (i.e. a distribution of likely places we might be) it will output a vector whose coördinates sum to 1, which corresponds to where we're likely to be after clicking on a random page with the probabilities given by the $rank(v_i)/\deg(v_i)$'s.

But is this really how we surf the internet? Not really: at any point in time, we've got some random chance of flying off to YouTube or Metafilter or something else. So: why don't we build that in? In other words, instead of looking at $A_G^T \cdot D_{\deg^+}^{-1}$, why not look at

$$\frac{\alpha}{n} \cdot J + (1-\alpha) \cdot A_G^T \cdot D_{\deg^+}^{-1},$$

where α is some small constant that measures the likelihood of us flying off to any other webpage at random[6]. If α is not very big and the internet has a lot of pages, this shouldn't change our rankings; basically, what we've done here is add very low-probability edges between all of the edges in our graph.

Yet, such a small change turns out to give us exactly what we want! Specifically, we have the following in linear algebra:

Theorem 3 *(Perron-Frobenius) If A is a **probability matrix** – i.e. the sum of the entries in every column of A is 1 – and every entry in A is positive, then A_G's largest eigenvalue is 1, and furthermore the eigenspace corresponding to this eigenvalue has dimension 1. In other words, for any such probability matrix, there is a unique corresponding ranking vector.*

In practice, we usually don't want to find this eigenvector explicitly, as the internet is kind of a big thing. Instead, we use the following other property of such "positive" probability matrices:

Theorem 4 *(Perron-Frobenius) If A is a **probability matrix** with all entries > 0 and $\mathbf{x}$ is any probability vector (i.e. any vector with nonnegative coördinates that sum to 1), then $A^n\mathbf{x}$ converges to A's ranking vector (i.e. the eigenvector for eigenvalue 1 whose coördinates sum to 1).*

So, if we want to find a ranking vector, it suffies to just run the above process a few times to get something that's "close enough" for alll practical intents and purposes (where our choice of α determines just how fast this convergence is, as smaller α-choices make our graph feel more "disconnected" and thus intuitively should make our convergence process take longer.)

[6] In practice, $\alpha = 15\%$ seems to be a pretty good constant for calculating this ranking vector quickly.