PROFESSOR STRANG:

So. Today's lecture is partly cleaning up some pieces left from 1-D problems. 3.1 was second order equations, 3.2 was fourth order equations for beams. And just a few more comments to make. But then I do want to say something about splines. And I included a homework question that asked you to use the spline command and get a result. I'm not planning to make that a serious topic in the course. In other words, you're not like going to be responsible on an exam for a discussion of splines. But it's such a major event, and major requirement in scientific computing -- if you're given some values, put a curve through them -- that I have to say something about that. So that's interpolation. Because you'll have this all the time. If your experiment produces some values at specific times or a finite number of values, and you want to fit a curve through those points. How do you do it? So splines give a very, very good answer.

And then, the next and big topic of the course we'll make a beginning on. Which is gradient, divergence, leading to Laplace's equation. Problems in 2-D. So partial differential equations are going to show up. OK.

Some small points. First point, about the MATLAB homework for Monday. A question came by email. I put the spike and also the jump in c, in the coefficient, not at mesh point. If you've started on that problem, you probably noticed that, and maybe a little swearing went on. [LAUGHTER] Because it makes it not so easy. And then the question came, did I really expect you to compute, use the exact position and figure out-- When you have some integrals to do, they'll change at the place where c changes values. Where we have integrals of c, if c has one of those jumps, on the quiz the jump came right at a neat point, so it was clear. c was one on one side and two on the other. Now the jump is going to come at an in-between point. So the answer is yes.

I had a nice email this morning from somebody who has done that MATLAB homework. And he said, quote, it's a good problem. I learned more from it than I did from the other problems in the pset. So I'll blame it on him. I'll stay with that problem, and ask you to do your best to

deal with the calculation. It's second order and I used linear elements. I didn't want to push you up to these cubic elements, even though those would give much better accuracy.

One reason I didn't put the spike and the jump right at the node is that, if I did, the finite element solution would be exactly right. Because the correct solution is going to be piecewise linear. And if I put the breaks between pieces right at node points, well, then I have a function that's in my finite element space, in my linear space. So it'll come out exactly. So I thought well, let's at least have some chance to see the error between the finite element solution and the exact one. You see what I'm saying? I'm just anticipating, that the exact solution will-- Let's see. Something happened at 1/3 and something happened at 2/3. I'm afraid I don't remember the boundary conditions. Does anybody remember what I took for boundary conditions? Probably free-fixed, or something. So maybe, if it was free-fixed, probably the solution would be maybe something like that. I don't know. Whatever. Piecewise linear with breaks where there's a jump in c, or where the point load is hitting. Yeah. I don't know if that picture is accurate.

But my point was, if I chose those points to be also mesh points, then the finite element solution would be exact, and we wouldn't learn anything about accuracy. OK. So anyway, for that MATLAB problem, I'm hoping you can do the integrals, which will mean noticing which integral-- If that doesn't fall, so if the mesh points are like this, in the finite element integral over that mesh, over that interval, you're going to have to split it into two pieces. That's it. So see if you can do that split, do the split into two pieces. OK. So that's a comment on that homework. OK.

What else do I need to comment on, when I'm sort of catching up here? Oh, about boundary conditions. I thought I would just repeat clearly on the board the rules about and the difference between fixed and free when we're doing this weak form approach to the problem. And then I gave the other names: a fixed condition, in this theory it's often called an essential boundary condition, but many people just name Dirichlet as the person's name that's associated with such conditions. Like u(0)=0. Right, fixed.

And then free conditions are natural conditions in the finite element method, and that means that we don't have to impose them. Right. I discussed this in the earlier lecture. I just thought I'd bring it back here, because it's easy to describe in one line.

Now somebody asked, on Wednesday, a good question. Suppose u' is given, but not given as

zero. So suppose we have a condition like $u'=A$, let's say. What to do? So I'll just put question mark. How do I deal with-- How do I? How does the weak form approach and the Galerkin approach deal with a non-zero, natural condition? A non-zero Neumann condition, that'd be the right word.

So this would be, in a second order problem, this would be a Neumann condition. And I'm thinking, what do you do if A isn't zero. Somehow that has to show up in your finite element equations, right? So I guess the right way is to go back to the way they came from. So you remember how the weak form came? I took the differential equation, I multiplied by any test function, and I integrated by parts. That's where the weak form started. So the integration by parts gave me this nice symmetric integral. It's symmetric in u and v. It only requires one derivative of u, so that I'm allowed to use hat functions. And I'm allowed to use hat functions for v, because their derivatives have just a jump, and a jump function I can integrate, no problem.

Now what about this $u'=A$? And I guess we're seeing it there. If u' was zero, if we had a totally free end, u' was zero, nothing happening there, then that term would drop out of the integration by parts. And you see why I don't have to impose anything on v, because that term is already accounted for by the u' going away.

Now, what about if u' is given, but not given zero? Suppose it's given the value A. All I want to say is, then this term has to stay. I have to pay attention to it. What happens when Galerkin takes over? Galerkin will put in one of his test functions. Maybe I just indicate that by changing the v to a cap V. It'll be one of them. There'll be n of them. Each V_i, each test guy, gives me an equation. And of course u is now going to-- The Galerkin idea is that instead of any u, I only have capital U's. Combinations of these phis.

I'm not going to say anything very big here, or very clear. All I want to say is that if u prime is given at an end point, and given by A, then this whole stuff is equaling something with an f, right? $f*V_i*dx$. I should remember the other side of the equation. If u' is given, then c at that end point, times the given value of A, times the V will be -- whatever value of V that is -- will show up in equation i. That will be a term that goes on the right-hand side. That's all I'm saying, and all you would expect. That any time I'm given some data, I'm given some non-zero boundary conditions, or I'm given some non-zero f in the inside, that stuff is going to show up on the right-hand side. It'll show up as part of the F. The vector of these. So up to now, F has just come from little $f(x)$, integrated against V. And all I'm saying is that if we had one of these

guys, then that would contribute to big F, also. Yeah. You can see, it's not beautiful. But we have to realize what we would do. OK, that's not my favorite topic. But I'll just say quit on that one. I'm not expecting you to do problems or anything that involved that possibility. Just to see that it could happen. OK. Quit.

Now, interpolation. All right. So I guess if I had to list problems that people face all the time and need numerical guidance on, this is one of them. And so let's say we're in 1-D -- 1-D is certainly a lot easier -- so, suppose I have this. I'll call this x, just to have a name for the variable. Suppose I have some values. Say six values. And I've measured them, I've worked hard to find those six values. But now you may say, well I'm thinking I have some function F(x) here. But right now I don't have a function. Right now I just have six values of that function. You know, what is the stretching constant, what is the c(x). If you had a physical experiment with an actual spring, you might measure, you might put on different forces and measure the stretching, and have six values of that.

How do I fit those with a curve? In other words, how do I know-- Interpolation means, inter-means asking, what about between the points? What value should it have there? OK, well there's one simple rule would be, just interpolate linearly between. OK. That's certainly pretty stable. It will not get out of control. But it's not very accurate. For the function that's probably lying behind this, this isn't that great of a representation.

OK, you say, I want something smoother. Well another idea that naturally comes up is, the other extreme would be-- So that was completely local. Just, every two values determine the broken line. The second idea that's completely natural would be, fit a polynomial. Fit a polynomial through those points, and then you have a nice, totally simple function. Nice curve. And so what degree polynomial would we be looking for here? If I have six points, I could fit a polynomial of degree-- What do you think? I want to have six coefficients, so I can get six equations to make the polynomial go through those six points. So my polynomial P(x), which actually goes through those points, would be some polynomial of the form, it's got some constant term, and it's got some linear term. And how far am I going to go? Fifth, right. That'll give me six coefficients. OK. So I could put a fifth degree polynomial, that gives me a_0, a_1, up to a_5. Six coefficients through six points.

Well, six isn't too bad. But if six became 60, don't do it. That's the message. Don't do it. If I'm going on up to, say I have an $a_{59} x$ to the 59th, And that fits my 60 points. And you say, well look, I've got 60 values, that should be way better than having only six. But the polynomial that

would come out of that, even if these values were pretty smooth, the polynomial that fits -- well I'd have to put in a whole lot more to get 60, and I won't try -- the polynomial would go crazy. Can I just draw something crazy? I mean, whatever. It's unstable.

And there are classical examples of that. In fact, the famous example is the function one over one plus x squared. And later, there's a figure in the book, I think it's in section 5.4, which shows the result. That's a terrific function. It's infinitely differentiable. I would even call it an analytic function. It's great. But if I tried to fit it with a high-degree polynomial at points, that polynomial gets out of hand. And a figure is in-- the figure there.

So, MATLAB or any computing system would have a subroutine that does interpolation. And this is named after Lagrange. So this is called Lagrange interpolation, fitting a polynomial. And if the degree is small, and maybe degree five would be okay, then that's an important thing to be able to do. In other words, you're finding these six coefficients from these six heights. You've got some six by six matrix that connects the six heights to the six a's. But when you get up to 60, that matrix stops behaving well. It becomes very ill conditioned, and your coefficients go all over the place.

OK, so my question is, what do you do? And one answer, one good answer is fit those points not with straight lines, that's too crude; not with a single polynomial, that's too unstable; but fit it with splines. So interpolation by splines. OK. So it's just sort of appearing in this section 3.2-- I mean by that, and people often do mean, when they say spline, they mean cubic spline. So you could have splines of other degrees. But often, sort of the natural choice is the cubic.

So can I just briefly describe what that does. What the cubic spline would be like. So let me draw in again these six points that I'm going to fit. And just say, for these few minutes, what's a cubic spline. I touched on that Wednesday, and now I just want to say a little bit more. And then the homework asked you to actually do it with values of a particular function. And see how close does the cubic spline come to the given function.

OK, so what's a cubic spline? What does that word, spline, what should that mean in your mind? So a cubic spline is a cubic in each piece. A cubic in each piece. Now we've met cubics in each piece as finite elements. And part of this short discussion is to keep those two slightly different ideas separate. The finite element idea was also piecewise cubic. But it used values and slopes. So it was completely local. If I had a value, as I have here, six values, and if I also had six slopes, if I knew the slopes, then I would use those cubic elements. And I would fit a--

If I knew the height and slope there, and I knew the height and slope there, there would be exactly one cubic. Because I would have four conditions, two conditions there, two conditions there, would be four. I could fit a cubic between, another cubic there, another cubic there. And because I'm using the same slope from the left and from the right, the slope would be good. It would be continuous. The second derivative, the curvature, would not be continuous with those cubic elements.

And that's the difference. So the difference is, splines have continuous, no jump in-- Let me just put it this way. No jump in the function. I'll use S, maybe, for spline. No jump in its slope. And no jump in the second derivative. So that's the difference. That the spline functions, the only jumps you see for those are jumps in the third derivative. So that makes this extremely smooth. So if I just try to draw now what a spline would do, it'll go through those points, coming out of the spline fit MATLAB command. And it'll be as smooth as I drew it. There is a change in the third derivative at these points.

Actually, have you ever seen this word, "spline," before? It came out of naval engineering. When people were designing the shape of the ship. A naval architect is fitting the proposed shape of a ship-- this was before MATLAB, yeah, before life started. [LAUGHTER] They had little physical, slightly bendable-- I'll call them splines. I guess that's what they were. That's where the word came from. Some physical thing which was like a curve that you use-- I don't know if you guys ever did mechanical drawing. That was a freshman subject when I came to MIT. Mechanical drawing. I was terrible, terrible, at mechanical drawing. I don't know. I had a friend who helped. I probably helped him in some other course. Anyway, whatever. So there were physical things, curves you used to use. And these spline curves were used, and maybe still are used, by naval architects in creating a drawing. But I'm assuming that those things are now all computerized, and the spline command is used. Anyway, the result is that.

Now, I have to draw one picture of a spline. Of the most important spline. And then I'm done with splines. So, what I'm going to draw is now a B-spline. And that's a basic spline. It could be one of our functions, it could be one of our functions phi(x). One of our trial functions. It could be, and let me comment on that. So let me remind myself, good or bad, with a question. And let me try to answer that. But let me first draw the B-spline.

OK, so it's like a hat function. Actually a hat function is a linear spline. A hat function is that low level spline that's linear between pieces. Now the B-spline is going to have the value one there. It's going to have no jump in the function. So the function will go through there. No jump

in the slope. No jump in the second derivative. And I want to get down to zero. I want to get down to zero. I want it to be as local as I can make it. So I want it to get to zero.

Here's the point. With those cubic finite elements, I got them down to zero. They came in here with zero slope, and then they continued as zero, no problem. If I'm wanting the second derivative also to be zero, to be continuous, I won't be able to do it in one interval. It's going to take me a total of four intervals. This one can come down to some point, here, where another cubic starts. Maybe I should do the one here. Going up, you remember the allowed cubic spline would be an x cubed over six, some multiple of x cubed. I don't know what multiple it'll take, maybe x cubed over six is right. It'll come up to some point here. Now I've got to get it beginning to curve downwards. So I'm going to have to change the third derivative. Change to another cubic there, change to another cubic there, and a final cubic here. So there's a picture of a B-spline. And we could figure out, by requiring all these continuity conditions, we could figure out the formula for the cubic in these four pieces. And it would be symmetric, of course, across that center point. But I think I won't try to do it. I'll just leave that idea, then. There's a figure in the book showing a picture of the B-spline.

So those are functions, extremely valuable in this interpolation problem. Because all splines are combinations of B-splines. Yeah, now let me pull this topic together. Every spline, every spline function, is combination of these B-splines. Let $B_i(x)$ be the one centered on node i. And then there's a neighbor centered on node i+1. And a neighbor to the left centered on node i-1. What's the point? The point is that, at a typical node, i+1, three of these functions will be non-zero. With these very local hat functions at that node, the only one of the phi functions that wasn't zero is the one I've drawn. All the others were zero there, right? All the other hats were zero. There was just the one.

But now there'll be a B-spline starting from here, going up to here, coming down from here. There'll be the B-spline starting from here, going up to here, up, down, so on. So at a typical node, I'm getting the one centered at that node, and also the one to the left, which is on its way down, and also the one to the right, which is on its way up. In other words, it's not as local as the other construction. So I would say good, because it's smooth, but bad because it's not fully local. Not completely local. It's reasonably local, in that there are only three functions that are affecting these points. If I use those polynomials, they weren't local at all. All the points, it was-- A typical x to the fifth is not zero at any of the points. These B-splines are zero at a lot of points, but three of them, one, two, and three, will be non-zero at a typical mesh point.

OK, so they're not popular, as a result, for finite elements. I'm just wanting to be sure you make the distinction. They're very popular for the interpolation job. They're very popular for the interpolation job. I've got some combination of these at particular mesh points. It's supposed to agree with F at those mesh points. This is my system to solve. I create these functions, these B-splines. I'm given F at typical points. And I choose a combination which matches F at those points. Yeah, OK.

Is there a question or discussion on this? I don't like to not say anything about such an important problem as putting curves through points. But I don't want to make it a course on splines. Yes?

AUDIENCE: [UNINTELLIGIBLE]

PROFESSOR STRANG: I don't know that I'll-- Oh, OK, yes. All right. Two or three comments, and that suggests a good question, a very good question.

Can I make a separate comment that if I go into two dimensions, this gets much tougher. What happens if I had a function of x and y? So that I've got points on a surface, and I'm trying to fit a surface to it. Just one message first: that's not as easy. It's pretty easy if those points are on a rectangular grid. So this is like typical. And then I'll come to your Nyquist question, which is a very good one. Can I just-- because we're coming in to 2-D now.

Suppose I-- There's two dimensions. I have a grid. Let's suppose it's a nice grid. And at every point, at every grid point, I have a height. And I'm fitting a surface. Right? My function is now a function of x and y. x is here, y is here, F is the surface coming out of board, in the third dimension. And fitting those points, if they're regularly spaced like that, my life would be OK. I could use sort of products of splines in 1-D. I could use products of basis-- of splines in the x-direction times splines in the y-direction. And it would be pretty successful. It would be, not quite as nice, but almost OK. But if I had irregularly spaced points from a general grid, it's not as easy. And I won't-- People have obviously had to figure out how to do it, and to repeat again, that's what the CAD/CAM world is having to do all the time, is fit a curve, fit a surface through points. It's significant.

Now, you asked about Nyquist. So that's a good question. Can I just say, I have to say what-- So what's a function called? Band-limited. How many have heard Nyquist's name? Okay, some of you may know a lot more than I about it. But let me just get some context for band-limited functions. Okay. This is actually a topic that will belong in the third part of our course, in

the Fourier part. So this is a Fourier idea. So I'll come back to it, actually. So right here I'm just going to say, very briefly, how it might connect us to what I've said today. But then let's make a plan to, when we've got the idea of Fourier coefficients.

What is a band-limited function? You know the Fourier idea is to take F(x), and write it as some combination of pure frequencies. $e^{(i*K*x)}$, let me say. $e^{(i*K*x)}$. So this is Fourier that's coming. I take the function F(x), and I write it-- I could think of it as a combination of pure exponentials, pure frequencies. That would be a Fourier series. So that's a Fourier series because I'm using-- K has integer values. The frequencies are zero, one, two, three, whatever. Now that we'll use, so that Fourier series comes for functions that are periodic. They repeat every 2pi. Because those functions, if I increase x by 2pi, don't change. So I'm repeating every 2pi.

Now I have to say a word about the other possibility, which would be to have all frequencies. I'll integrate now, dK. Instead of summing on K, I'll integrate on K. What does band-limited mean? Band-limited means that only frequencies in a certain range, say a range around zero, are included. So a band-limited function would be a function whose frequencies go from some value, say minus omega to omega, instead of going from minus infinity to infinity. This would be band-limited frequencies between minus omega and omega. So that's another kind of smooth function. That's another way-- Functions that have only low frequencies are associated with smoothness. High frequencies are associated with fast oscillations. So the class of band-limited functions gives me another, a Fourier way, to talk about smoothness. So for us, smoothness was something about how many derivatives were continuous. That's the sort of smoothness in the x domain. How many derivatives. Smoothness in the frequency domain is, how fast do the frequencies drop off. And here, band-limited means they drop like a shot. Band-limited means that the frequencies in the function, that these $e^{(i*K*x)}$'s are not there for high frequencies. High frequencies are out for a band-limited function.

And then Shannon has a formula for the natural way to fit-- So our same interpolation problem. So now, completing the answer to your question. So I have these points. So Shannon could fit a function, and it would look smooth, through those points. And his function would be band-limited. So it would be smooth again. It would be-- yeah, it would be smooth. In some way, this Shannon band-limited stuff is the limit of splines as the spline degree goes way up. So we did hat functions, degree one splines. Cubic splines I recommended as a pretty reliable construction. But you could do fifth degrees splines, seventh degree splines, you could

keep going. And in the limit, you would get this. So maybe that's some partial answer to the connection between splines and this Fourier world. OK. Thanks.

So these are topics now-- Wow, today's lecture is kind of-- Can I do one really important thing now in today's lecture? For you to remember? Gradient and divergence? I don't want you to spend the weekend without thinking about gradient and divergence. [LAUGHTER] OK.

Here's the idea. For lots and lots of applications, for a region, let's say, in the plane, I have the same-- What physical example shall I pick now? Maybe I'll let u be the temperature. So instead of being displacement, let me make it temperature. u. OK. Then I have a temperature gradient. A slope. But now, the whole point is, that's a function of x and y. Then I have a temperature gradient. And if I'm consistent with the notation, that'll be e(x,y). And then you'll expect that there's some c(x,y). Some operator c that tells me how much heat flows. This will tell me something about the the, the thermal conductivity. Right?

Really, as I speak about this framework, I'm just uttering the correct words. Having started with temperature, this thing should be a temperature gradient. Then I should have some physical thermal conductivity, different for different metals or different materials. And I'll have a heat flow, w(x,y). And everybody knows that w will be c times e. And then there will be some A transpose. Of course, there will be some A transpose here, and some A here. And up here I'll have a balance equation. OK.

I just want to think, what's the operator A? If we can focus on that question, then that's what's going to occupy us for, certainly the whole of next week. So I've actually used the word gradient. We have functions of two variables. We're looking for the change, the rate of change, the steepness of those functions. So this A, Au, is going to give me two derivatives. I've got two variables, there are two first derivatives. Both of them are important. That's what the A is. For the next big example in the course. The final major example of the course is, when A acts on a function of two variables, because I'm in a region in the plane, to find its rate of change. And this is called the gradient. The shorthand is the gradient of u.

So we have to understand that. We have to understand what the gradient is. And, of course, we want to know its transpose. So can I just think, what should be the transpose of the gradient? OK. I'll take that picture out. OK. Thinking now about the transpose of the gradient.

OK. So A itself is, you could say, is d/dx and d/dy. You notice how I'm separating out A from Au. When this acts on a function, this is the gradient operator that acts on a function, u, to

produce du/dx and du/dy. OK. Now what's the transpose of this? OK. You can guess what it should be, and then we'll see that yes, that guess is correct. So what should A transpose look like? If there's any justice, A transpose should be-- OK, this is two by one. The transpose you would expect to be a row, a row vector. I should have the transpose of that there. And what is the transpose of that? Just tell me. Because we have an idea. What should be the transpose of d/dx? Negative d/dx. And what should be the transpose of d/dy? It should be negative d/dy. Those are two different pieces. This is not run together. There's a big space in there. That's two pieces.

So what is A transpose applied to a-- So, heat flow w. I want to say, what is A transpose applied to w? You're going to see this again, but we'll just take these minutes to show it for the first time. So A transpose-- Wait a minute. Is w a function, or is it a vector? Yeah we've got to get that straight before we start. Here's an ordinary function, a scalar function. Just whatever, x squared plus y squared. What is e?

Suppose this is x squared plus y squared. Let's have a specific example. What would e then be? It's got two components, right? It's got an x derivative and a y derivative. e is a vector, [2x, 2y]. Then I multiply by c. So this w has two components. This has got a $w_1(x,y)$, $w_2(x,y)$. So just keep things straight.

So it's that that I want to apply A transpose to. So A transpose w is minus d/dx, minus d/dy. And everything's coming out right because it's applied to a function w_1 and w_2. w is a vector field. It's not a scalar field, it's a vector field. And the result is just what it should be: minus dw_1/dx minus dw_2/dy. OK, good.

This has got to come out of integration by parts, right? So we'll have to think: what does integration by parts mean in two variables? And it's a famous formula named after Gauss and Green. The Green's formula, often. But do you recognize what I'm looking at here? This is so important it has a name. And what's the name? And then we're ready to go. What's the name of, if I take a vector field, like [2x, 2y]. Let me take [2x, 2y]. as a specific example. Suppose w is [2x,2y]. That was multiplied by, let me take c to be one. Then what is A transpose w? Specifically. What am I getting out of it? What do I get from here? That's minus the x derivative of the first guy, and the x derivative of that guy is two, so I'm getting a minus two. And this is minus the y derivative of the second guy, so that's another minus two. So I'm getting a number. It happens to be a number here. I chose such a simple function it came out to be a number. And what's the name? So this is minus the what of w? Just tell me, what's the name

everybody uses for that operation? The divergence. Minus the divergence of w. So I what I'm saying here is that this-- I'm saying it because somehow I remember studying vector calculus. And in that process, I learned about the gradient, and I learned about the divergence. But I never learned that one was the transpose of the other. I think, looking back, that was criminal. To describe those-- With a minus sign, of course. I learned Green's formula, but now we'll see what it means. OK, that's next week's job. Have a great weekend and see you Monday.