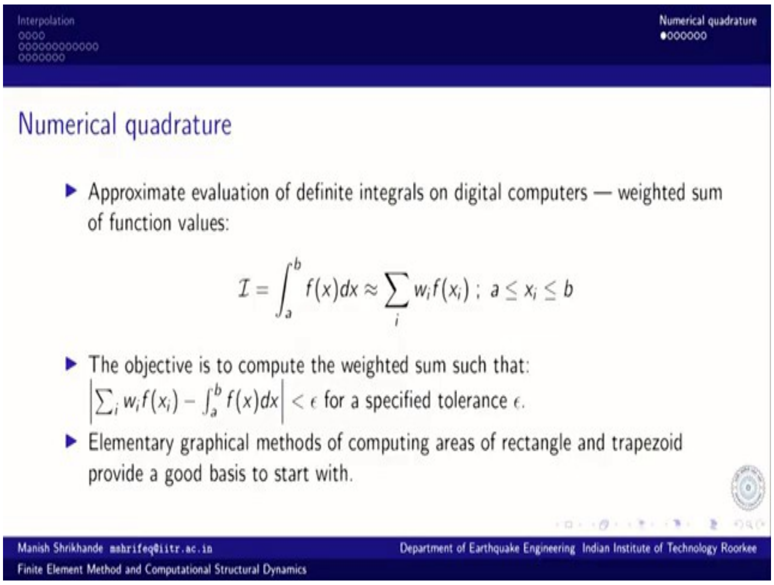
Finite Element Method and Computational Structural Dynamics
Prof. Manish Shrikhande
Department of Earthquake Engineering
Indian Institute of Technology, Roorkee

Lecture - 07
Polynomial Interpolation and Numerical Quadrature - IV

As I mentioned earlier, the Finite Element operations have to be performed on digital computers. Integrations over definite interval on digital computers are approximations and are called numerical quadrature schemes.

Analytical evaluation of integrals is very slow and very often may not be possible to have analytical expression that can be integrated even with symbolic computer system that might be available. So, it is not always possible to use analytical integration.

(Refer Slide Time: 01:22)



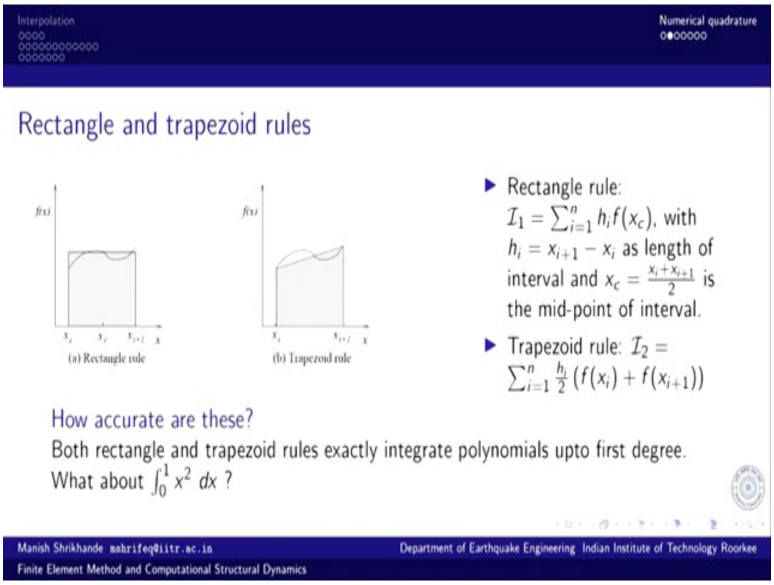
And it is much more convenient and much more efficient to use numerical integration. It is very quick, very efficient and accurate enough for our purposes. And the way definite integrals are evaluated on a digital computers or using any quadrature rule are essentially based on weighted sum of function values. So, we have an integral with f(x) as integrand and that has to be integrated between the range a to b.

The integral is evaluated as a weighted sum of function values. Function f is evaluated at several points x_i in the range a to b, and there are some weights assigned based on the sampling points, and those weights are multiplied with the function values at the points and then they are added up. And this particular summation, weighted summation is taken as an approximation of the definite integral.

It is simple enough as you can see it is only accumulation. Addition is the simplest operation on digital computer. So, it is a very simple scheme per se. And the objective is to compute the weighted sum such that the error is under certain specified threshold. So, the weighted sum that we have is within certain tolerance level of the exact integral that might be there for any function.

In the elementary graphical methods of computing that we all studied, the areas under any curve, are essentially integrals. So, for example, rectangle rule or trapezoid rule, they are the goods starting points for studying numerical quadrature.

(Refer Slide Time: 03:58)



So, we take the first on the left hand side is what we call as rectangle rule, on the right hand side we have what we call as trapezoid rule. In rectangle rule we find out the value of the function at the midpoint of the interval. So, x_c is the centroid, that is the central point in this interval x_i and x_{i+1}.

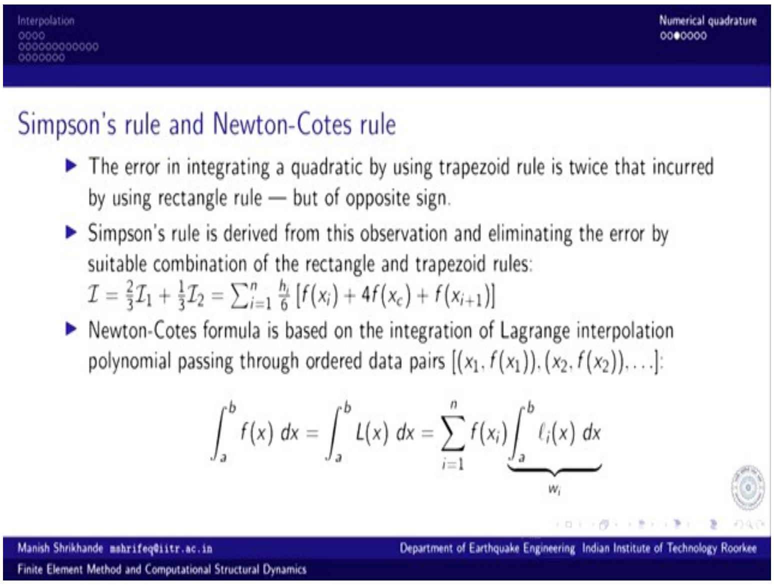
We find out the function value at x_c and assume that constant function value holds over the entire interval (x_i to x_{i+1}) and then we compute the area of rectangle. So, there will be some areas where it would be overestimated and there would be some areas, some regions, where it will be underestimated than the true sample. But, if we take this interval small enough then it should average out and results should be satisfactory.

In trapezoidal rule, we take the function values at the end points. So, if x_i and x_{i+1} are the sampling points, we just take the function values at x_i and function value at x_{i+1}, and then just use the area of trapezoid between these two points. And that is taken as an approximation for the area under the curve between x_i to x_{i+1}.

Now, both these methods rectangle rule as well as trapezoidal rule are first order accurate, and error decreases linearly as we reduce the interval between adjacent points. In rectangle rule, the coefficient (weight) is length of the interval h. So, for i^{th} interval between x_i and x_{i+1}, if the interval is h_i, then h_i becomes the weighting coefficient and the function is evaluated at the centre of this interval. And then, obviously, it is just weighted sum of these function values at the middle of the intervals. In trapezoid rule, the coefficient is of course, $h_i/2$, and we take the function values at the ends of the interval. This is essentially taking average of the function values at the two end points and multiplying it by the interval gives the area of this trapezoid.

Now, interestingly if we try to figure out what is the error; then we find that trapezoid rule incurs slightly higher error than rectangle rule. The approximation error will be incurred only when we try to integrate a second degree polynomial by these two rules. So, let us try to integrate x^2 by using both of these - rectangle rule and the trapezoid rule, over the interval 0 to 1, just to serve as an example.

(Refer Slide Time: 07:55)



So, you can check it out the error in integrating a quadratic by using trapezoid rule is two times that incurred using rectangle rule. Work it out and you will be surprised at this result. But interestingly the error is of opposite sign in both the methods. And therein lies opportunity and this is how numerical algorithms are. The scope of improvement in numerical algorithms are detected. How do we make use of fundamental results, fundamental observations and use that to our advantage?

So, two basic first order methods have an error of the same order. They are first order accurate, but the error term is of opposite signs. So, that means, if we can somehow add them together then the error term will vanish because the error is of opposite sign. Also we now know that error in trapezoid rule is two times that of the rectangle rule.

So, if I use the rectangle rule and to double that, and added to the trapezoid rule, then the error term will vanish, and I will get a higher degree of accuracy. And that is how Simpson's rule is derived. Simpson's rule is derived from this observation and eliminating the error by a suitable combination of rectangle and trapezoid rules.

If you recall I_1 refers to rectangle rule and I_2 refers to the trapezoid rule. So, two-third weightage is given to rectangle rule result and one-third weightage is given to trapezoid rule result, and resulting summation is familiar Simpson's rule and that has a higher degree of accuracy, it is second order accurate.

If we go further in this context, then we can develop a series of formula, Newton-Cotes formula based on integration of Lagrange interpolation polynomials. So, if we have sampling points and the function values and do not have the analytical expression then we can actually construct a Lagrangian interpolation.

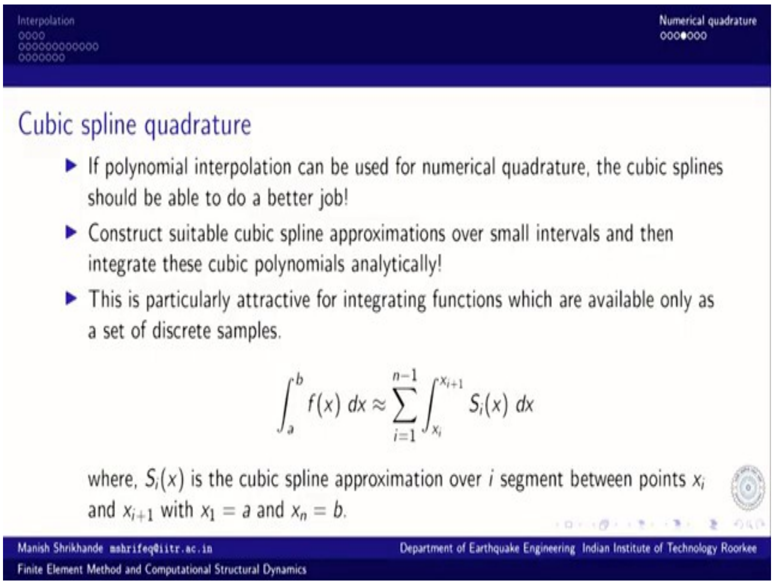
And once we have the Lagrangian interpolation, the integral of Lagrangian interpolation is a well known function and the integral can be evaluated and the data sets can be populated and that is how the Newton-Cotes formulas are derived. So, weighting coefficients in the weighted sum, are essentially the terms related to integration of Lagrangian interpolation polynomials between the range of integration.

Another interesting issue is these are all based on polynomial interpolation and very handy. But if polynomial interpolation is not very good at approximating the function over the entire range, cubic spline interpolation can provide a very good approximation to the function over a smaller range.

Hence we can divide the range into smaller segments, and construct cubic spline over smaller segments. Once we have the cubic splines we have an analytical expression which we can simply integrate and get the results for numerical integration - definite integral over the certain range.

So, the process, obviously begins with construction of suitable cubic spline over smaller intervals and then integrate those cubic polynomials analytically and nothing can possibly go wrong here.

(Refer Slide Time: 12:53)



This is particularly attractive for integrating functions which are available only as a set of discreet samples. So, if the function values are only available as a set of ordered pair of points then I can very easily construct cubic splines and then integrate the cubic spline functions analytically over individual segments and that can be taken as approximation for integrating over the tabulated data points between certain range a to b.

Now, all these numerical quadratures that we have seen so far, have a common theme - definite integral is replaced or is approximated by a weighted sum of function values. So, function has to be evaluated at specified points unless it is given as a data pair.

But if it is not available, it has to be evaluated at specific points, then there is cost involved, computational cost involved in computation, in evaluating the function at specified points. Function evaluation is the most expensive operation in this whole process because sometimes very complex functions need to be evaluated. There are so many things that go on in evaluating the function particularly in finite element approximation that we will develop.

So, it is always worthwhile to explore which method or which integration method will provide sufficient accuracy with minimum number of function evaluations. And it so happens that Gauss-Legendre quadrature as they are called, are the optimal in terms of sampling required for integrating polynomials.

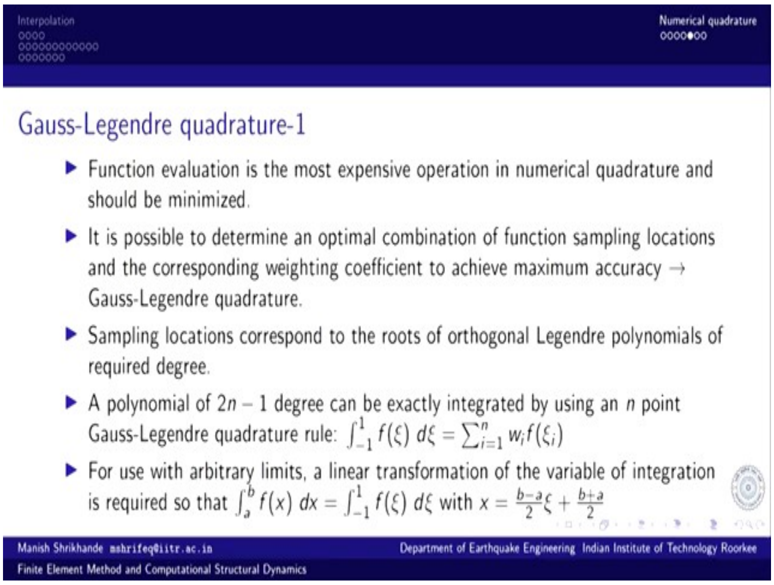
As long as we are integrating polynomials and over a finite range, there is the minimum number of function evaluations that we may need for a specified degree of accuracy. The Gaussian quadrature or Gauss-Legendre quadrature as it is called is the most efficient computing technique.

So, what is done? I mean, obviously, the basic forms still remains the same function integral. Definite integral is still evaluated as weighted sum of function values. However, the sampling points are determined based on the optimality criteria.

We try to find out what is the most accurate approximation that can be constructed. Most accurate evaluation of the integral for a given number of sampling points and find out what should be the optimal location of those sampling points and associated with what should be the optimal weighting coefficient. So, both the combinations are left to the optimization process, the weighting coefficient as well as the sampling point.

So, we try to find out what is the weighting coefficient and what is the corresponding sampling point to achieve the maximum accuracy in the numerical quadrature.

(Refer Slide Time: 17:08)



The derivation is a little bit involved and not really worth spending time on it. What we really need is the result. It so happens that the optimal sampling locations correspond to the roots of orthogonal Legendre polynomials of required degree. This theory of polynomials has a very interesting contributions to the numerical computations.

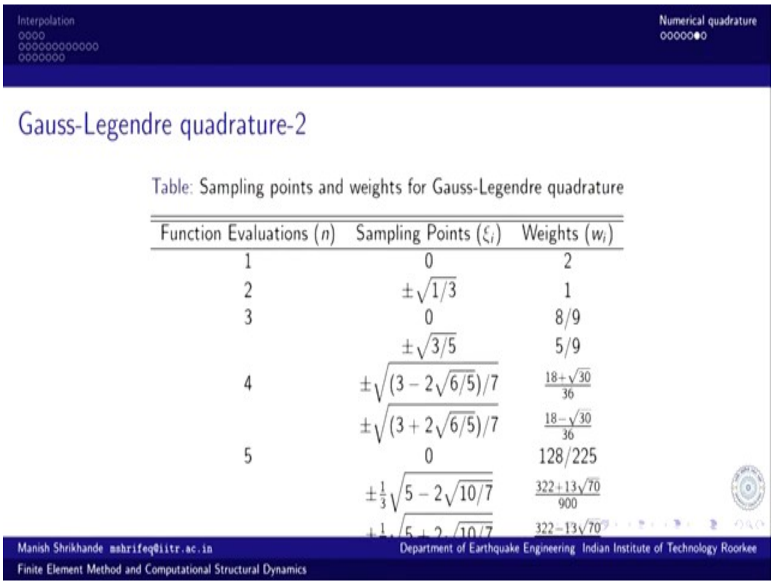
So, as we saw roots of orthogonal Chebyshev polynomials giving us optimal locations for function polynomial interpolations. Now, we have roots of orthogonal Legendre polynomials which would give the optimal locations for evaluation of numerical quadrature.

So, a polynomial of (2n - 1) degree can be exactly integrated by using a n point Gauss-Legendre quadrature defined over interval of -1 to +1. Any arbitrary interval can be mapped into this range and function values by suitable change of variables and the integral is evaluated by weighted sum of function values over the interval.

So, a n point quadrature rule would have these n number of terms w_i multiplied by function evaluated at ξ, ξ being the sampling point location which would correspond to the root of Legendre polynomial over the range -1 to +1.

And for use with arbitrary limits, a linear transformation between the variables of integration can be imposed, so that integral between a to b can be mapped to integral between -1 to +1, with suitable change of variable, so x can be mapped to normalized variable ξ by suitable linear transformation.

(Refer Slide Time: 19:31)



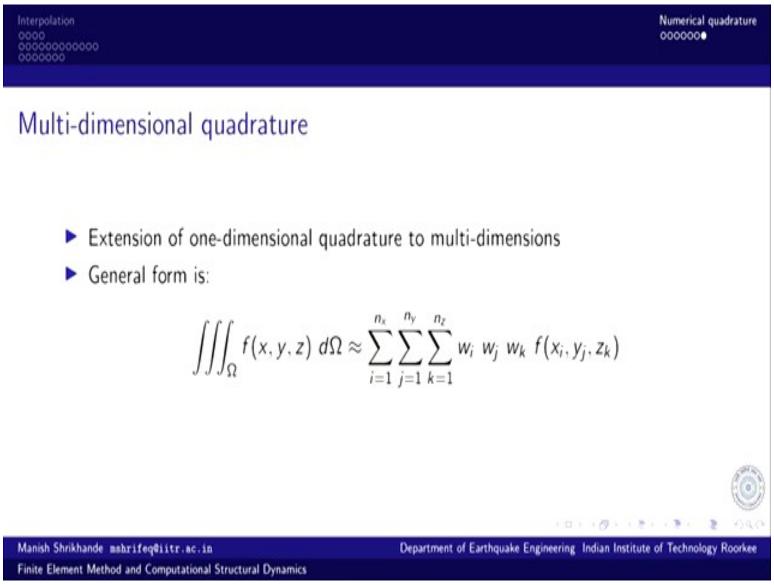
So, typically we have these sampling points and associated weights as given in table. If we have only one function evaluation, single point evaluation, then that corresponds to the midpoint of the domain [-1, +1] and the weight is 2. And polynomial of order (2n - 1) can be exactly integrated. So, first degree polynomial can be exactly integrated and as you can see that corresponds to rectangle rule if you work this out.

If we go by two number of function evaluations the sampling points are $+1/\sqrt{3}$ and $-1/\sqrt{3}$ and weighting coefficient is unity. And for 2 number of function evaluation the highest degree of polynomial that can be integrated exactly is 3. i.e. we can integrate cubic polynomial exactly by just 2 function evaluations.

And if I go to 3 number of function evaluations then I can integrate up to fifth degree polynomial exactly. And the sampling points are 0 with weight of 8/9 and $\pm\sqrt{3/5}$ with the weight of 5/9.

In finite element approximation we will rarely have need to go beyond 3 point quadrature rule, so that actually suffices for all our practical purposes. But many books on numerical analysis will give you appropriate sampling points and weights for different orders of Gauss-Legendre quadrature that can be used for analysis and most efficient evaluation of numerical quadrature.

(Refer Slide Time: 22:16)



Now, that was for one-dimension when there was only one independent variable. What to do in case of two-dimensions or three-dimensions? Our problems can be multi-dimensional. So, the new integration also has to be evaluated with respect to several dimensions - double integrals and triple integrals.

The general form is very easily extended to multi-dimensional integral where each integral is replaced by a weighted sum. So, a triple integral is replaced by a triple sum, approximated by a triple sum, and weight and the sampling points are chosen appropriately just in the same way as in the case of one-dimensional evaluation. So, with this we can compute the numerical integral as we may need during the process of our operations of finite element method formulation, and then that will set the stage for further calculations.

So, with this we wrap up the basic introduction and basic tools that we may need for development of finite element approximation. We will begin with discussion of finite element formulation from the next lecture.

Thank you.