Title: Explorations in Numerical Relativity (PHYS 642) - Lecture 1

Date: Mar 15, 2010 11:20 AM

URL: http://pirsa.org/10030100

Abstract:

Explorations in Gravitational Physics–Numerical Relativity

Matt Choptuik, UBC
Luis Lehner, Perimeter/Guelph
Frans Pretorius, Princeton

Perimeter Institute
Waterloo, ON
March 15–April 2, 2010

http://bh0.phas.ubc.ca/pi-nr/www/

Course Outline

- Solution of Classical Field Equations Using Finite Difference Techniques (Matt)
  1. Solving the wave equation using finite difference techniques
  2. $3+1$ approach to the Einstein equations
  3. Dynamical spherically symmetric spacetimes
  4. Spherically symmetric Einstein-Klein-Gordon Evolution
  5. Introduction to Black Hole Critical Phenomena

- General Relativistic Hydrodynamics Using Gudonov/HRSC Schemes (Luis)
  1. Mathematical structure; Linearly degenerate vs truly nonlinear eqns
  2. Burgers eqn; Godunov Methods & the Riemann problem
  3. $3+1$ Approach to GRHydrodynamics
  4. Stationary solutions, TOV stars & perturbations
  5. Magnetohydrodynamics & miscellaneous topics

- Topics in Numerical Relativity (Frans)
  1. Gravitational waves overview (nature in GR & sources)
  2. Newman Penrose formalism, Teukolsky equation
  3. BSSN/generalized harmonic evolution
  4. Adaptive mesh refinement (AMR)/parallel computation

Week 1: References

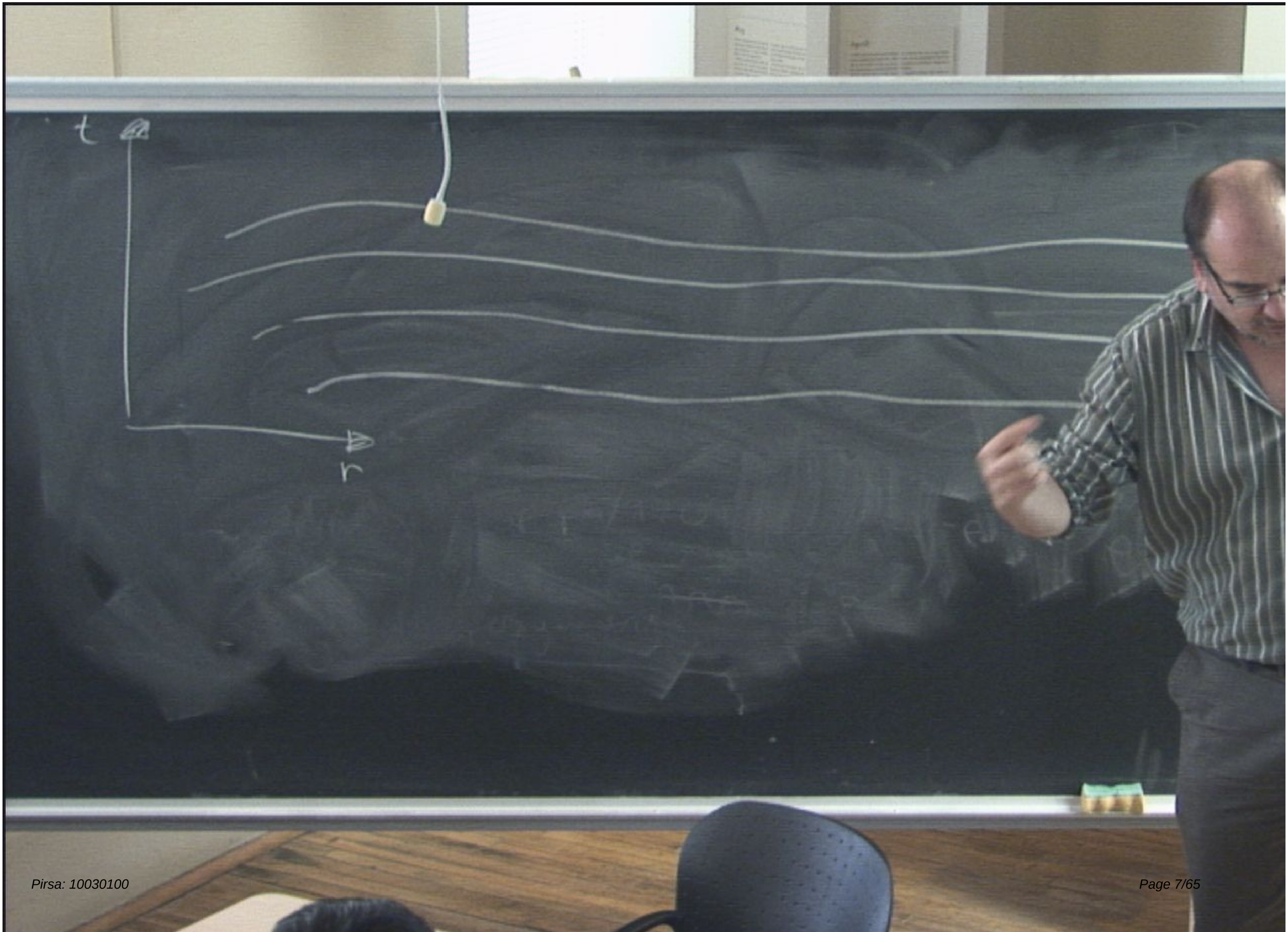
- Mitchell, A. R., and D. F. Griffiths, **The Finite Difference Method in Partial Differential Equations**, New York: Wiley (1980)
- Richtmeyer, R. D., and Morton, K. W., **Difference Methods for Initial-Value Problems**, New York: Interscience (1967)
- H.-O. Kreiss and J. Oliger, **Methods for the Approximate Solution of Time Dependent Problems**, GARP Publications Series No. 10, (1973)
- Gustatsson, B., H. Kreiss and J. Oliger, **Time-Dependent Problems and Difference Methods**, New York: Wiley (1995)

Solution of Classical Field Equations Using Finite Difference Techniques

1. Solving the wave equation using finite difference techniques

Preliminaries

- Classical field equations $\equiv$ time dependent partial differential equations (PDEs)
- Can divide time-dependent PDEs into two broad classes:
  1. **Initial-value Problems (Cauchy Problems)**, spatial domain has no boundaries (either infinite or "closed"—e.g. "periodic boundary conditions")
  2. **Initial-Boundary-Value Problems**, spatial domain *finite*, need to specify boundary conditions
- **Note:** Even if *physical* problem is really of type 1, finite computational resources $\longrightarrow$ finite spatial domain $\longrightarrow$ approximate as type 2; will hereafter loosely refer to either type as an IVP.
- *Working Definition:* **Initial Value Problem**
  - State of physical system arbitrarily (usually) specified at some initial time $t = t_0$.
  - Solution exists for $t \geq t_0$; uniquely determined by equations of motion (EOM) and boundary conditions (BCs).


t
r

Preliminaries

- Classical field equations ≡ time dependent partial differential equations (PDEs)
- Can divide time-dependent PDEs into two broad classes:
  1. **Initial-value Problems (Cauchy Problems)**, spatial domain has no boundaries (either infinite or "closed"—e.g. "periodic boundary conditions")
  2. **Initial-Boundary-Value Problems**, spatial domain *finite*, need to specify boundary conditions
- **Note:** Even if *physical* problem is really of type 1, finite computational resources ⟶ finite spatial domain ⟶ approximate as type 2; will hereafter loosely refer to either type as an IVP.
- *Working Definition:* **Initial Value Problem**
  - State of physical system arbitrarily (usually) specified at some initial time $t = t_0$.
  - Solution exists for $t \geq t_0$; uniquely determined by equations of motion (EOM) and boundary conditions (BCs).

Preliminaries

- Approximate solution of initial value problems using *any* numerical method, including finite differencing, will always involve three key steps
  1. Complete mathematical specification of system of PDEs, including boundary conditions and initial conditions
  2. Discretization of the system: replacement of continuous domain by discrete domain, and approximation of differential equations by algebraic equations for discrete unknowns
  3. Solution of discrete algebraic equations

- Will assume that the set of PDEs has a unique solution for given initial conditions and boundary conditions, and that the solution does not "blow up" in time, unless such blow up is expected from the physics

- Whenever this last condition holds for an initial value problem, we say that the problem is well posed

- Note that this is a non-trivial issue in general relativity, since there are in practice *many* distinct forms the PDEs can take for a given physical scenario (in principle infinitely many), and not all will be well-posed in general

Preliminaries

- Classical field equations $\equiv$ time dependent partial differential equations (PDEs)
- Can divide time-dependent PDEs into two broad classes:
  1. **Initial-value Problems (Cauchy Problems)**, spatial domain has no boundaries (either infinite or "closed"—e.g. "periodic boundary conditions")
  2. **Initial-Boundary-Value Problems**, spatial domain *finite*, need to specify boundary conditions
- **Note:** Even if *physical* problem is really of type 1, finite computational resources $\longrightarrow$ finite spatial domain $\longrightarrow$ approximate as type 2; will hereafter loosely refer to either type as an IVP.
- *Working Definition:* **Initial Value Problem**
  - State of physical system arbitrarily (usually) specified at some initial time $t = t_0$.
  - Solution exists for $t \geq t_0$; uniquely determined by equations of motion (EOM) and boundary conditions (BCs).

Preliminaries

- Approximate solution of initial value problems using *any* numerical method, including finite differencing, will always involve three key steps

  1. Complete mathematical specification of system of PDEs, including boundary conditions and initial conditions
  2. Discretization of the system: replacement of continuous domain by discrete domain, and approximation of differential equations by algebraic equations for discrete unknowns
  3. Solution of discrete algebraic equations

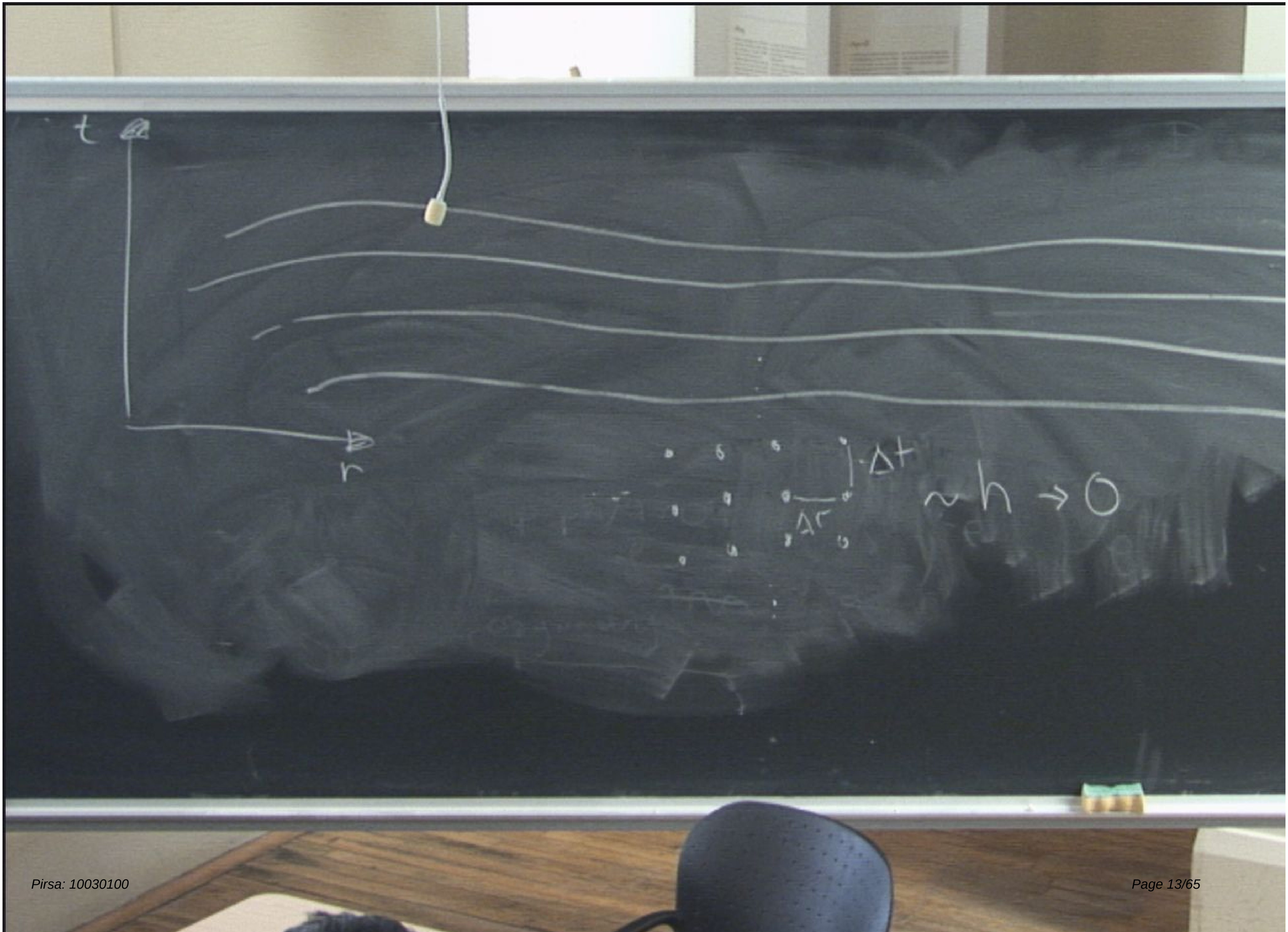
- Will assume that the set of PDEs has a unique solution for given initial conditions and boundary conditions, and that the solution does not "blow up" in time, unless such blow up is expected from the physics

- Whenever this last condition holds for an initial value problem, we say that the problem is well posed

- Note that this is a non-trivial issue in general relativity, since there are in practice *many* distinct forms the PDEs can take for a given physical scenario (in principle infinitely many), and not all will be well-posed in general

Preliminaries

- Approximate solution of initial value problems using *any* numerical method, including finite differencing, will always involve three key steps
  1. Complete mathematical specification of system of PDEs, including boundary conditions and initial conditions
  2. Discretization of the system: replacement of continuous domain by discrete domain, and approximation of differential equations by algebraic equations for discrete unknowns
  3. Solution of discrete algebraic equations
- Will assume that the set of PDEs has a unique solution for given initial conditions and boundary conditions, and that the solution does not "blow up" in time, unless such blow up is expected from the physics
- Whenever this last condition holds for an initial value problem, we say that the problem is well posed
- Note that this is a non-trivial issue in general relativity, since there are in practice *many* distinct forms the PDEs can take for a given physical scenario (in principle infinitely many), and not all will be well-posed in general


t
r
Δt
Δr
~ h → 0

Preliminaries

- Approximate solution of initial value problems using *any* numerical method, including finite differencing, will always involve three key steps

  1. Complete mathematical specification of system of PDEs, including boundary conditions and initial conditions
  2. Discretization of the system: replacement of continuous domain by discrete domain, and approximation of differential equations by algebraic equations for discrete unknowns
  3. Solution of discrete algebraic equations

- Will assume that the set of PDEs has a unique solution for given initial conditions and boundary conditions, and that the solution does not "blow up" in time, unless such blow up is expected from the physics

- Whenever this last condition holds for an initial value problem, we say that the problem is well posed

- Note that this is a non-trivial issue in general relativity, since there are in practice *many* distinct forms the PDEs can take for a given physical scenario (in principle infinitely many), and not all will be well-posed in general

Why Finite Differencing?

- There are several general approaches to the numerical solution of time dependent PDEs, including
  1. Finite differences
  2. Finite volume
  3. Finite elements
  4. Spectral
- Finite difference (FD) methods are particularly appropriate when the solution is expected to be smooth (infinitely differentiable) given that the initial data is smooth
- This is the case for many classical field theories including those for a scalar (linear/nonlinear Klein Gordon), vector (electromagnetism [Maxwell]), rank-2 symmetric tensor (general relativity [Einstein])
- In cases where solutions do *not* remain smooth, even if the initial data is—as happens in compressible hydrodynamics, for example, where shocks can form—the finite volume approach is the method of choice (next week)

Why Finite Differencing?

- Accessibility: Requires a minimum of mathematical background: if you're mathematically mature enough to understand the nature of the PDEs you need to solve, you're mathematically mature enough to understand finite differencing

- Flexibility: Technique can be used for essentially any system of PDEs that has smooth solutions, irrespective of
  - Number of dependent variables (unknown functions)
  - Number of independent variables (a.k.a. "dimensionality" of the system: nomenclature "1-D" means dependence on one spatial dimension plus time, "2-D", "3-D" similarly mean dependence on two/three dimensions, plus time, respectively)
  - Nonlinearity
  - Form of equations: technique does not require that the system of equations has any particular/special form (contrast with finite volume methods where one generally wants to cast the equations in so-called conservation-law form)

Why Finite Differencing?

- Accessibility: Requires a minimum of mathematical background: if you're mathematically mature enough to understand the nature of the PDEs you need to solve, you're mathematically mature enough to understand finite differencing

- Flexibility: Technique can be used for essentially any system of PDEs that has smooth solutions, irrespective of
  - Number of dependent variables (unknown functions)
  - Number of independent variables (a.k.a. "dimensionality" of the system: nomenclature "1-D" means dependence on one spatial dimension plus time, "2-D", "3-D" similarly mean dependence on two/three dimensions, plus time, respectively)
  - Nonlinearity
  - Form of equations: technique does not require that the system of equations has any particular/special form (contrast with finite volume methods where one generally wants to cast the equations in so-called conservation-law form)

Why Finite Differencing?

- Error analysis:
  - Mathematically rigorous: Quite difficult
  - Practical/empirical: Extremely straightforward—basic principle is to compute multiple solutions using same initial data and problem parameters, but differing fundamental discretization scales. Comparison of solutions provides direct estimate of error in solutions
- Adaptivity: Can combine basic method with changes in
  - Local scale of discretization
  - Order of approximation

  in order to maximize increase in solution accuracy as a function of computational work invested (e.g. adaptive mesh refinement, week 3)
- Parallelization: Due to "locality of influence" in finite difference schemes, it is relatively easy to write FD codes than run efficiently on large distributed memory computer clusters having 1000s or cores (these days 10,000s or even 100,000s!)

Why Finite Differencing?

- Sufficiency: FD techniques are often sufficient to generate solutions of acceptable accuracy, again assuming that solutions are smooth
  - Will usually not be the most efficient and/or accurate among possible approaches, but when one is looking for a solution for the first time (science vs engineering/technology), such considerations are often not very important
- Now proceed to illustration of finite difference technique through the solution of the simple and familiar 1-D wave equation

The 1-D Wave Equation

- Consider the following initial value (Cauchy) problem for the scalar function $\phi(t, x)$

$$\phi_{tt} = c^2 \phi_{xx}\,, \quad -\infty \le x \le \infty\,, \quad t \ge 0 \tag{1}$$

$$\phi(0, x) = \phi_0(x) \tag{2}$$

$$\phi_t(0, x) = \Pi_0(x) \tag{3}$$

  where c is a positive constant, we have adopted the subscript notation for partial differentiation, e.g. $\phi_{tt} \equiv \partial^2\phi/\partial t^2$, and we wish to determine $\phi(t, x)$ in the solution domain from the initial conditions (2–3) and the governing equation (1)

- Note the following:
  - Since the spatial domain is unbounded, there are *no* boundary conditions
  - Since the equation is second order in time, two functions-worth of initial data must be specified: the initial scalar field profile, $\phi_0(x)$, and the initial time derivative, $\Pi_0(x)$
  - This system is well posed, and if the initial conditions $\phi_0(x)$ and $\Pi_0(x)$ are smooth—which we will hereafter assume—so is the complete solution $\phi(t, x)$

The 1-D Wave Equation

- Eqn. (1) is a *hyperbolic* PDE, and as such, its solutions generically describe the propagation of disturbances at some finite speed(s), which in this case is c
- Without loss of generality, we can assume that we have adopted units in which this speed satisfies $c = 1$. Our problem then becomes

$$\phi_{tt} = \phi_{xx}\,, \quad -\infty \le x \le \infty\,, \quad t \ge 0 \tag{4}$$

$$\phi(0,x) = \phi_0(x) \tag{5}$$

$$\phi_t(0,x) = \Pi_0(x) \tag{6}$$

- In the study of the solutions of hyperbolic PDEs, using either closed form (preferred to "analytic") or numerical approaches, the concept of characteristic is crucial
- Loosely, in a spacetime diagram, characteristics are the lines/surfaces along which information/signals propagate(s).

The 1-D Wave Equation

- Eqn. (1) is a *hyperbolic* PDE, and as such, its solutions generically describe the propagation of disturbances at some finite speed(s), which in this case is c

- Without loss of generality, we can assume that we have adopted units in which this speed satisfies $c = 1$. Our problem then becomes

$$\phi_{tt} = \phi_{xx}\,, \quad -\infty \le x \le \infty\,, \quad t \ge 0 \tag{4}$$

$$\phi(0,x) = \phi_0(x) \tag{5}$$

$$\phi_t(0,x) = \Pi_0(x) \tag{6}$$

- In the study of the solutions of hyperbolic PDEs, using either closed form (preferred to "analytic") or numerical approaches, the concept of characteristic is crucial

- Loosely, in a spacetime diagram, characteristics are the lines/surfaces along which information/signals propagate(s).

The 1-D Wave Equation

- Eqn. (1) is a *hyperbolic* PDE, and as such, its solutions generically describe the propagation of disturbances at some finite speed(s), which in this case is c

- Without loss of generality, we can assume that we have adopted units in which this speed satisfies $c = 1$. Our problem then becomes

$$\phi_{tt} = \phi_{xx}\,, \quad -\infty \le x \le \infty\,, \quad t \ge 0 \tag{4}$$

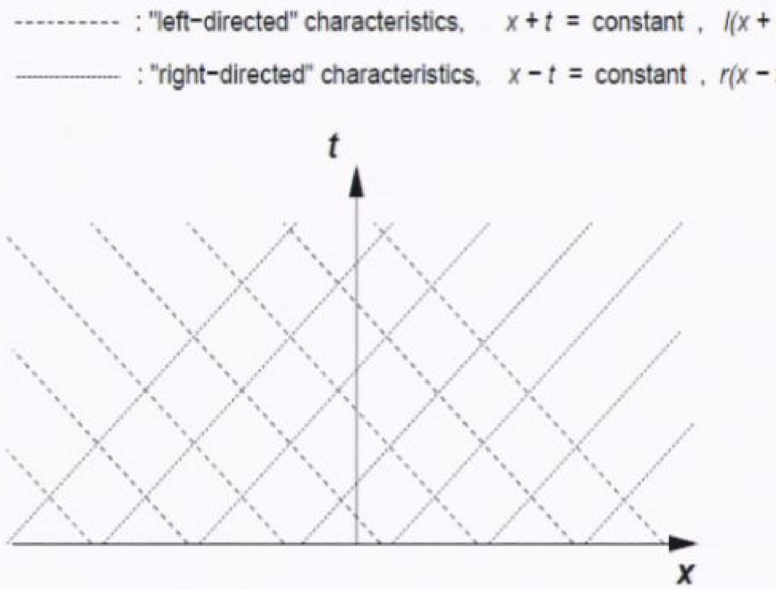
$$\phi(0,x) = \phi_0(x) \tag{5}$$

$$\phi_t(0,x) = \Pi_0(x) \tag{6}$$

- In the study of the solutions of hyperbolic PDEs, using either closed form (preferred to "analytic") or numerical approaches, the concept of characteristic is crucial

- Loosely, in a spacetime diagram, characteristics are the lines/surfaces along which information/signals propagate(s).

The 1-D Wave Equation



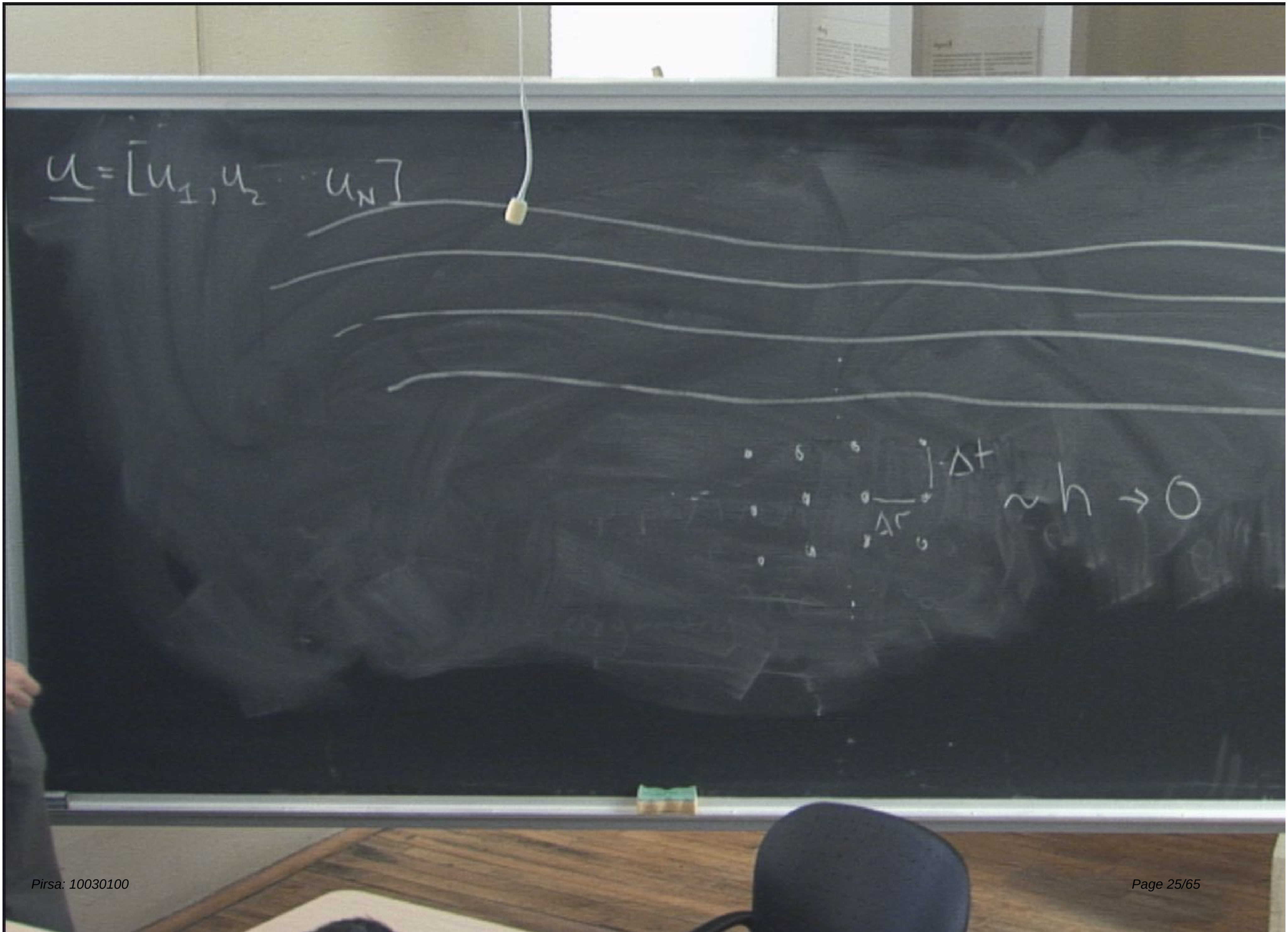
- General solution of (4) is a superposition of an arbitrary *left-moving* profile ($v = -c = -1$), and an arbitrary *right-moving* profile ($v = +c = +1$); i.e.

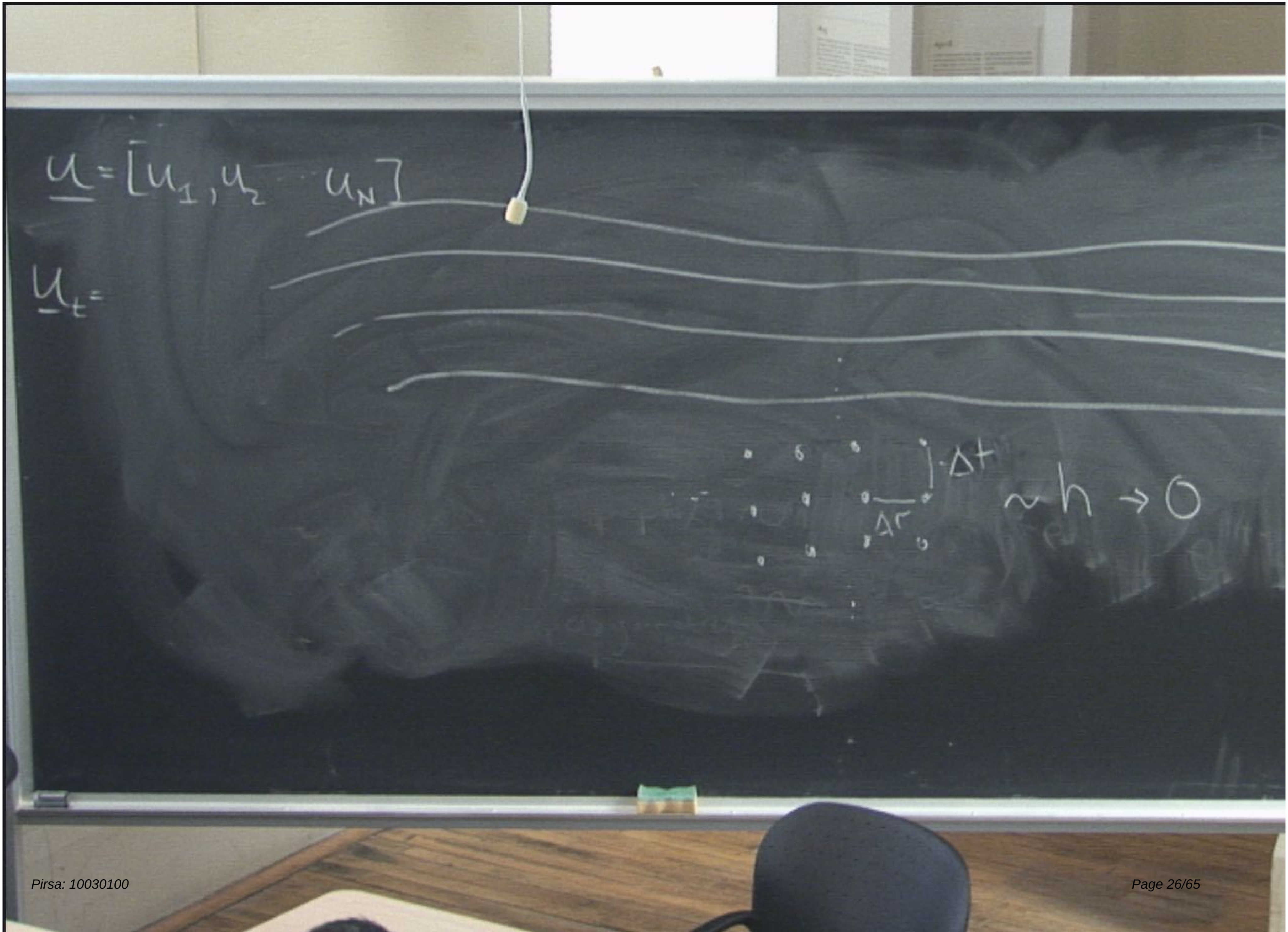
$$\phi(t,x) = \ell(x+t) + r(x-t) \tag{7}$$

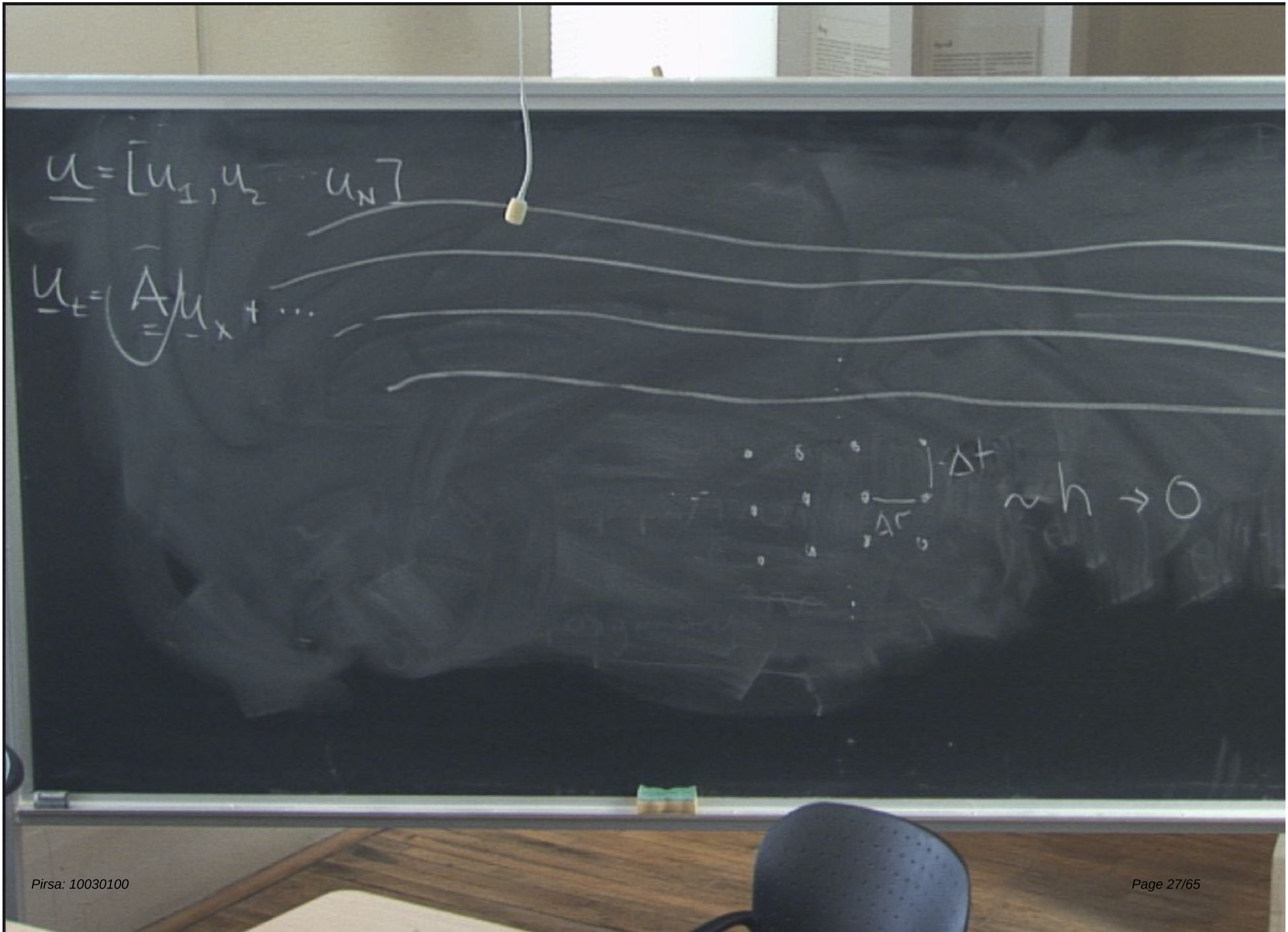
where

ℓ : constant along "left-directed" characteristics

r : constant along "right-directed" characteristics


u = [u1, u2 ... uN]
j·Δt
Δr
~ h → 0


u = [u1, u2 ... uN]
ut =
Δt
Δr
~ h → 0


u = [u1, u2 ... uN]
ut = A ux + ...
Δt
Δr
~ h → 0

The 1-D Wave Equation

- Observation provides alternative way of specifying initial values—often convenient in practice

- Rather than specifying $u(x,0)$ and $u_t(x,0)$ directly, specify *initial* left-moving and right-moving parts of the solution, $\ell(x)$ and $r(x)$

- Specifically, set

$$\phi(x,0) = \ell(x) + r(x) \tag{8}$$

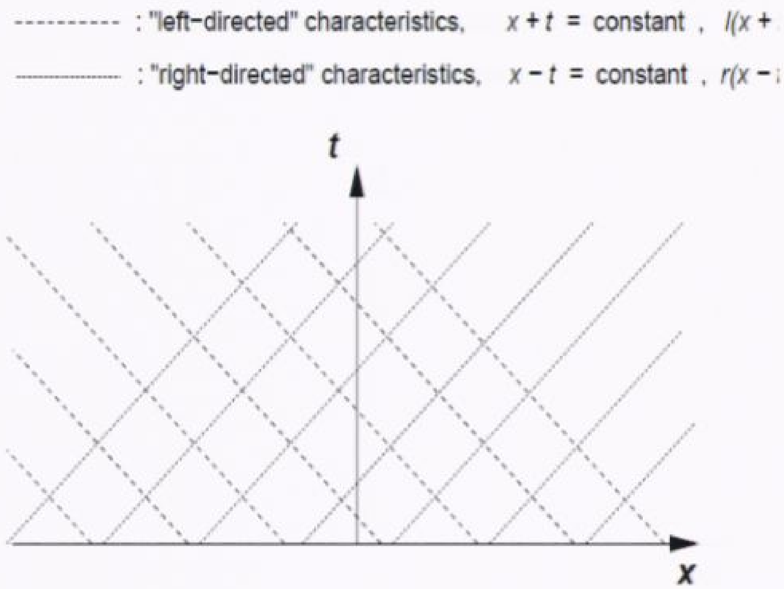
$$\phi_t(x,0) = \ell'(x) - r'(x) \equiv \frac{d\ell}{dx}(x) - \frac{dr}{dx}(x) \tag{9}$$

- For illustrative purposes will frequently take profile functions $\phi_0(x)$, $\ell(x)$, $r(x)$ to be "gaussians", e.g.

$$\phi_0(x) = A \exp\left[-\left(\left(x - x_0\right)/\delta\right)^2\right] \tag{10}$$

where A, x_0 and δ are viewed as adjustable parameters that control the overall

The 1-D Wave Equation



- General solution of (4) is a superposition of an arbitrary *left-moving* profile ($v = -c = -1$), and an arbitrary *right-moving* profile ($v = +c = +1$); i.e.

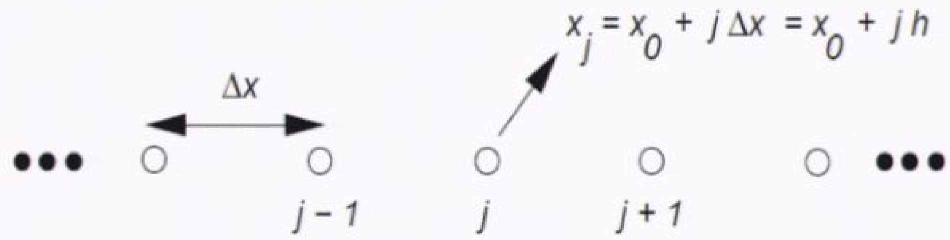
$$\phi(t, x) = \ell(x + t) + r(x - t) \qquad (7)$$

where

ℓ : constant along "left-directed" characteristics

r : constant along "right-directed" characteristics

Deriving Finite Difference Formulae



- One-dimensional, uniform finite difference mesh.
- Note that the spacing, $\Delta x = h$, between adjacent mesh points is *constant*.
- Will tacitly assume that the origin, x_0, of coordinate system is $x_0 = 0$.

Deriving Finite Difference Formulae

- Given the three values $u(x_j - h), u(x_j)$ and $u(x_j + h)$, denoted u_{j-1}, u_j, and u_{j+1} respectively, can compute an $O(h^2)$ approximation to $u_x(x_j) \equiv (u_x)_j$ as follows

- Taylor expanding, have

$$
\begin{aligned}
u_{j-1} &= u_j - h(u_x)_j + \frac{1}{2}h^2(u_{xx})_j - \frac{1}{6}h^3(u_{xxx})_j + \frac{1}{24}h^4(u_{xxxx})_j + O(h^5) \\
u_j &= u_j \\
u_{j+1} &= u_j + h(u_x)_j + \frac{1}{2}h^2(u_{xx})_j + \frac{1}{6}h^3(u_{xxx})_j + \frac{1}{24}h^4(u_{xxxx})_j + O(h^5)
\end{aligned}
$$

- Now seek a linear combination of u_{j-1}, u_j, and u_{j+1} which yields $(u_x)_j$ to $O(h^2)$ accuracy, i.e. we seek c_-, c_0 and c_+ such that

$$c_-\, u_{j-1} + c_0\, u_j + c_+\, u_{j+1} = (u_x)_j + O(h^2)$$

Deriving Finite Difference Formulae

- Results in a system of three linear equations for u_{j-1}, u_j, and u_{j+1}:

$$\begin{aligned} c_- + c_0 + c_+ &= 0 \\ -hc_- + hc_+ &= 1 \\ \frac{1}{2}h^2 c_- + \frac{1}{2}h^2 c_+ &= 0 \end{aligned}$$

which has the solution

$$\begin{aligned} c_- &= -\frac{1}{2h} \\ c_0 &= 0 \\ c_+ &= +\frac{1}{2h} \end{aligned}$$

- Thus, $O(h^2)$ FDA (finite difference approximation) for the first derivative is

$$\frac{u(x+h) - u(x-h)}{2h} = u_x(x) + O(h^2) \qquad (11)$$

Deriving Finite Difference Formulae

- Given the three values $u(x_j - h), u(x_j)$ and $u(x_j + h)$, denoted u_{j-1}, u_j, and u_{j+1} respectively, can compute an $O(h^2)$ approximation to $u_x(x_j) \equiv (u_x)_j$ as follows

- Taylor expanding, have

$$
\begin{aligned}
u_{j-1} &= u_j - h(u_x)_j + \frac{1}{2}h^2(u_{xx})_j - \frac{1}{6}h^3(u_{xxx})_j + \frac{1}{24}h^4(u_{xxxx})_j + O(h^5) \\
u_j &= u_j \\
u_{j+1} &= u_j + h(u_x)_j + \frac{1}{2}h^2(u_{xx})_j + \frac{1}{6}h^3(u_{xxx})_j + \frac{1}{24}h^4(u_{xxxx})_j + O(h^5)
\end{aligned}
$$

- Now seek a linear combination of u_{j-1}, u_j, and u_{j+1} which yields $(u_x)_j$ to $O(h^2)$ accuracy, i.e. we seek c_-, c_0 and c_+ such that

$$
c_-\, u_{j-1} + c_0\, u_j + c_+\, u_{j+1} = (u_x)_j + O(h^2)
$$

Deriving Finite Difference Formulae

- Given the three values $u(x_j - h), u(x_j)$ and $u(x_j + h)$, denoted u_{j-1}, u_j, and

which can be eliminated using 2 values (namely $u(x+h)$ and $u(x-h)$) suggests that the error might be $O(h)$.

- Fact that the $O(h)$ term "drops out" a consequence of the *symmetry*, or *centering* of the stencil: common theme in such FDA, called *centred* difference approximations

- Using same technique, can easily generate $O(h^2)$ expression for the *second* derivative, which uses the same difference stencil as the above approximation for the first derivative.

$$\frac{u(x+h) - 2u(x) + u(x-h)}{h^2} = u_{xx}(x) + O(h^2) \qquad (12)$$

- *Exercise:* Compute the precise form of the $O(h^2)$ terms in expressions (11) and (12).

Deriving Finite Difference Formulae

- May not be obvious *a priori*, that the truncation error of approximation *is* $O(h^2)$
- Naive consideration of the number of terms in the Taylor series expansion which can be eliminated using 2 values (namely $u(x+h)$ and $u(x-h)$) suggests that the error might be $O(h)$.
- Fact that the $O(h)$ term "drops out" a consequence of the *symmetry*, or *centering* of the stencil: common theme in such FDA, called *centred* difference approximations
- Using same technique, can easily generate $O(h^2)$ expression for the *second* derivative, which uses the same difference stencil as the above approximation for the first derivative.

$$\frac{u(x+h) - 2u(x) + u(x-h)}{h^2} = u_{xx}(x) + O(h^2) \qquad (12)$$

- *Exercise:* Compute the precise form of the $O(h^2)$ terms in expressions (11) and (12).

Sample FDA for the 1-D Wave Equation

- Let us consider the 1-D wave equation again, but this time on the finite spatial domain, $0 \le x \le 1$, where we will prescribe fixed (Dirichlet) boundary conditions
- Then we wish to solve

$$\begin{aligned} \phi_{tt} &= \phi_{xx} \qquad (c=1) \qquad 0 \le x \le 1, \quad t \ge 0 \qquad (13) \\ \phi(0,x) &= \phi_0(x) \\ \phi_t(0,x) &= \Pi_0(x) \\ \phi(t,0) &= \phi(t,1) = 0 \qquad (14) \end{aligned}$$

- We will again require that the initial data functions, $\phi_0(x)$ and $\Pi_0(x)$ be smooth
- Moreover, in order to ensure a smooth solution everywhere, the initial values must be compatible with the boundary conditions, i.e.

$$\phi_0(0) = \phi_0(1) = \Pi_0(0) = \Pi_0(1) = 0 \qquad (15)$$

Sample FDA for the 1-D Wave Equation

- Let us consider the 1-D wave equation again, but this time on the finite spatial domain, $0 \le x \le 1$, where we will prescribe fixed (Dirichlet) boundary conditions

- Then we wish to solve

$$\begin{aligned} \phi_{tt} &= \phi_{xx} \qquad (c=1) \qquad 0 \le x \le 1, \quad t \ge 0 \qquad (13) \\ \phi(0,x) &= \phi_0(x) \\ \phi_t(0,x) &= \Pi_0(x) \\ \phi(t,0) &= \phi(t,1) = 0 \qquad (14) \end{aligned}$$

- We will again require that the initial data functions, $\phi_0(x)$ and $\Pi_0(x)$ be smooth

- Moreover, in order to ensure a smooth solution everywhere, the initial values must be compatible with the boundary conditions, i.e.

$$\phi_0(0) = \phi_0(1) = \Pi_0(0) = \Pi_0(1) = 0 \qquad (15)$$

Sample FDA for the 1-D Wave Equation

- Let us consider the 1-D wave equation again, but this time on the finite spatial domain, $0 \le x \le 1$, where we will prescribe fixed (Dirichlet) boundary conditions

- Then we wish to solve

$$\begin{aligned} \phi_{tt} &= \phi_{xx} \qquad (c=1) \qquad 0 \le x \le 1, \quad t \ge 0 & (13) \\ \phi(0,x) &= \phi_0(x) \\ \phi_t(0,x) &= \Pi_0(x) \\ \phi(t,0) &= \phi(t,1) = 0 & (14) \end{aligned}$$

- We will again require that the initial data functions, $\phi_0(x)$ and $\Pi_0(x)$ be smooth

- Moreover, in order to ensure a smooth solution everywhere, the initial values must be compatible with the boundary conditions, i.e.

$$\phi_0(0) = \phi_0(1) = \Pi_0(0) = \Pi_0(1) = 0 \qquad (15)$$

Sample FDA for the 1-D Wave Equation

- Let us consider the 1-D wave equation again, but this time on the finite spatial domain, $0 \le x \le 1$, where we will prescribe fixed (Dirichlet) boundary conditions

- Then we wish to solve

$$\begin{aligned} \phi_{tt} &= \phi_{xx} \qquad (c=1) \qquad 0 \le x \le 1, \quad t \ge 0 & (13) \\ \phi(0,x) &= \phi_0(x) \\ \phi_t(0,x) &= \Pi_0(x) \\ \phi(t,0) &= \phi(t,1) = 0 & (14) \end{aligned}$$

- We will again require that the initial data functions, $\phi_0(x)$ and $\Pi_0(x)$ be smooth

- Moreover, in order to ensure a smooth solution everywhere, the initial values must be compatible with the boundary conditions, i.e.

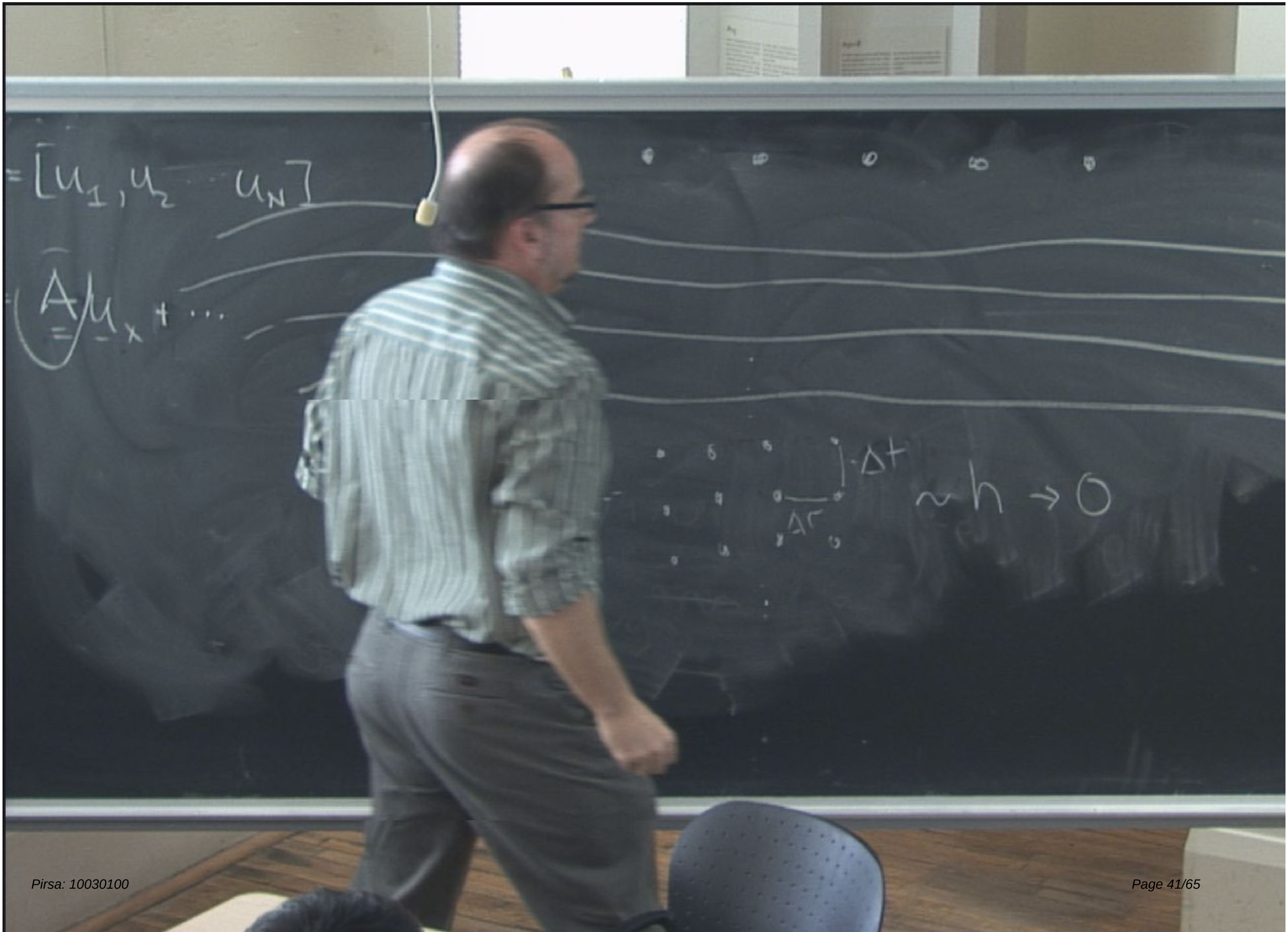
$$\phi_0(0) = \phi_0(1) = \Pi_0(0) = \Pi_0(1) = 0 \qquad (15)$$

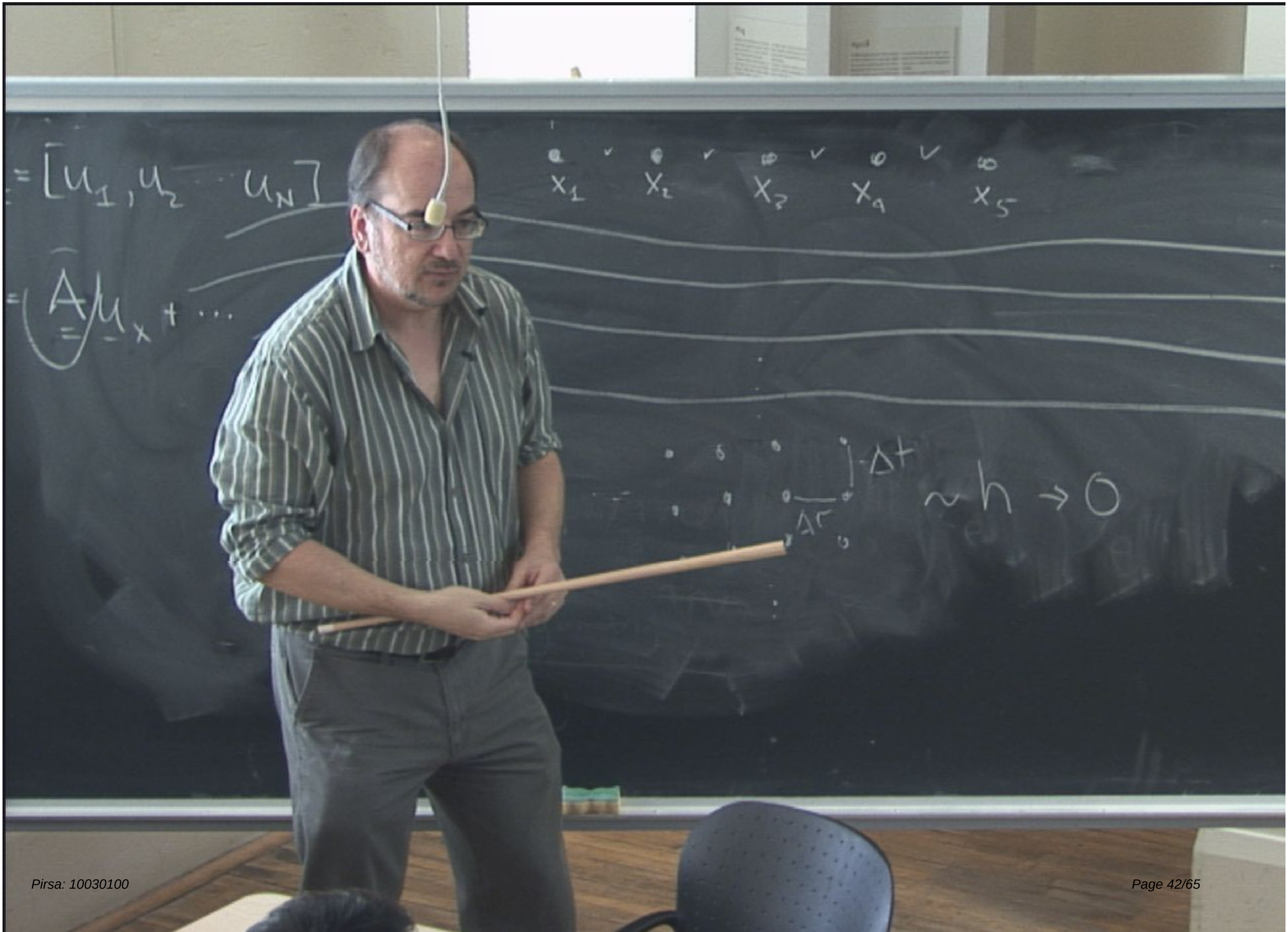
Sample FDA for the 1-D Wave Equation

- As always, we begin the discretization process by replacing the continuum solution domain with a finite difference mesh, whose typical element (point/event) we will denote by (x_j, t^n):

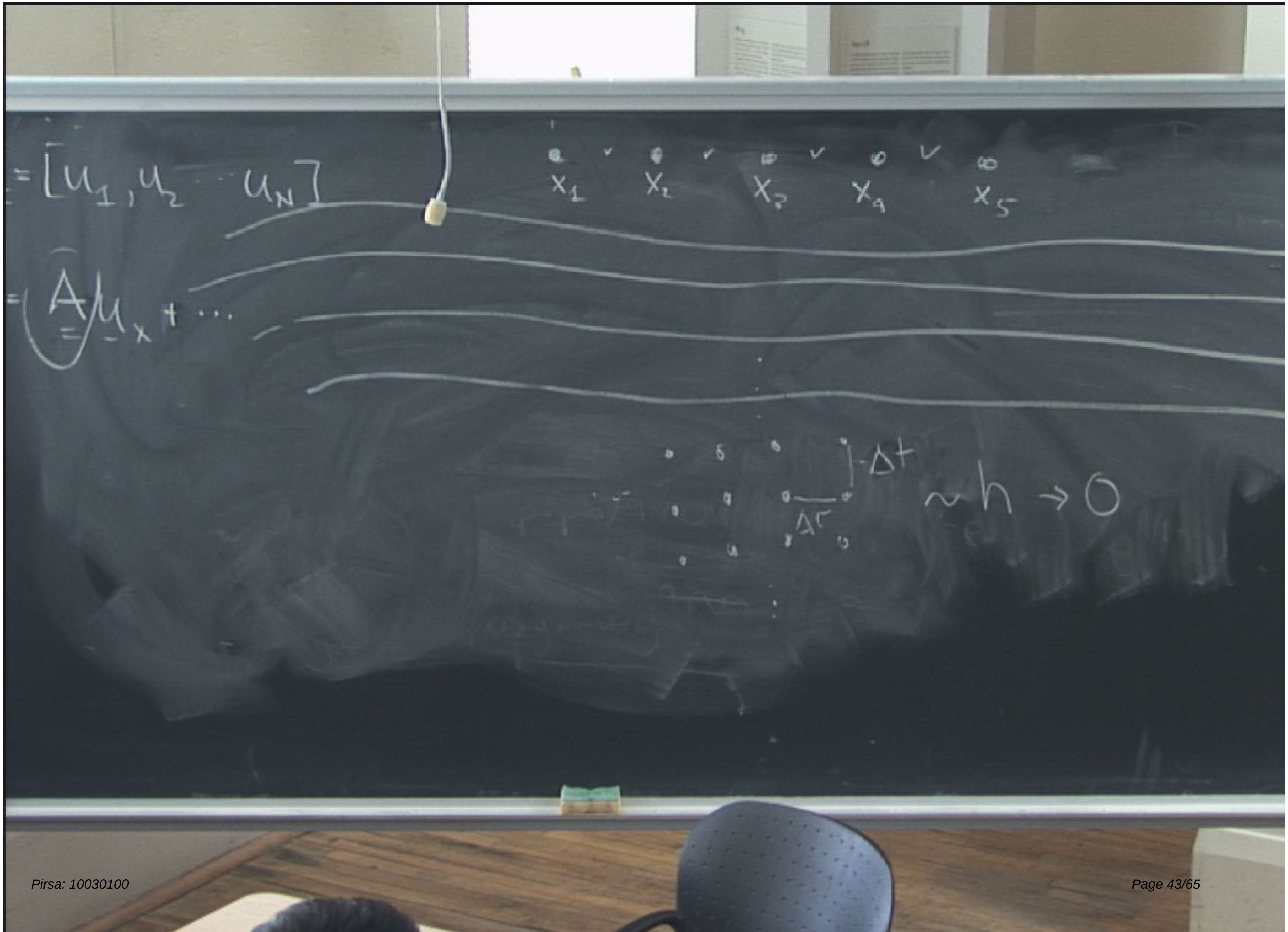
$$
\begin{aligned}
t^n &\equiv n\,\triangle t\,, \qquad n = 0, 1, 2, \cdots \\
x_j &\equiv (j-1)\,\triangle x\,, \qquad j = 1, 2, \cdots J \\
\phi^n_j &\equiv \phi(n\,\triangle t\,, (j-1)\,\triangle x\,) \\
\triangle x &= (J-1)^{-1} \\
\triangle t &= \lambda\,\triangle x \qquad \lambda \equiv \text{“Courant number”}
\end{aligned}
$$

- We note in passing that the quantity λ defined above is often called the Courant number or Courant factor, after the great 20th century mathematician Richard Courant who was a pioneer in the study of finite difference solutions of time dependent PDEs (in particular, in the use of FD techniques to establish existence and uniqueness of such PDEs))


$= [u_1, u_2 \cdots u_N]$
Δt
Δr
$\sim h \to 0$



$= [u_1, u_2 \cdots u_N]$
$= \underline{\underline{A}} \underline{u}_x + \cdots$
x_1 x_2 x_3 x_4 x_5
Δt
Δr
$\sim h \to 0$


= [u1, u2 ... uN]
x1 x2 x3 x4 x5
= A ux + ...
Δt
Δr
~ h → 0

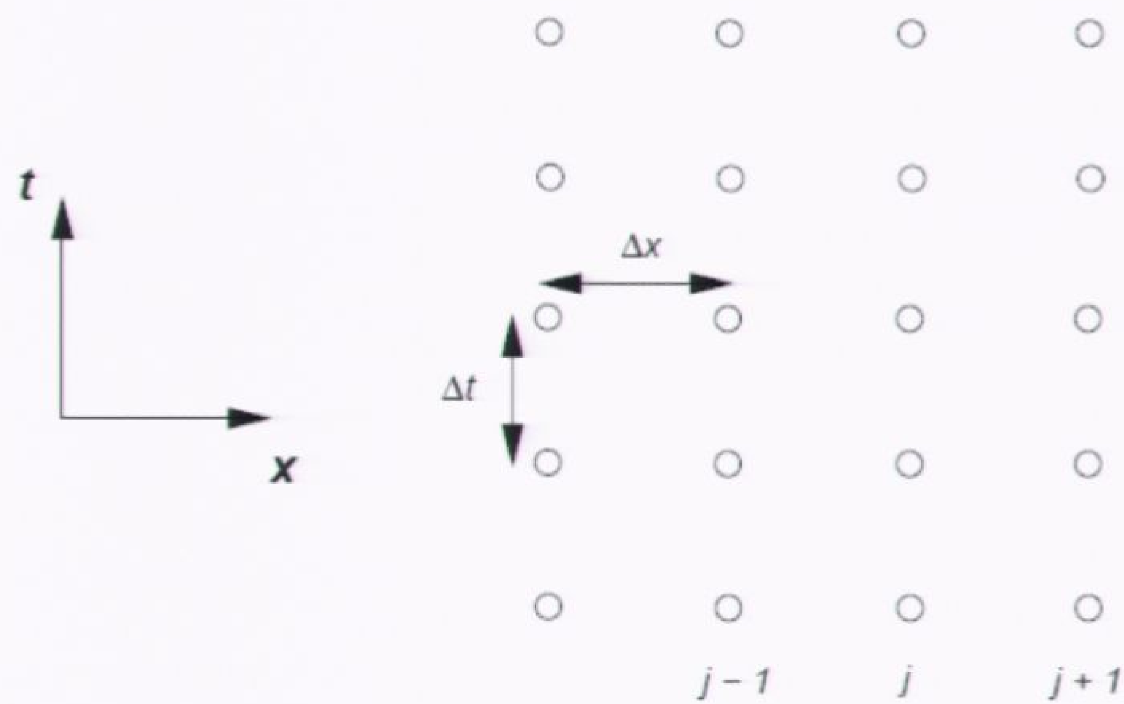
Sample FDA for the 1-D Wave Equation

- As always, we begin the discretization process by replacing the continuum solution domain with a finite difference mesh, whose typical element (point/event) we will denote by (x_j, t^n):

$$
\begin{aligned}
t^n &\equiv n\,\triangle t\,, \qquad n = 0, 1, 2, \cdots \\
x_j &\equiv (j-1)\,\triangle x\,, \qquad j = 1, 2, \cdots J \\
\phi^n_j &\equiv \phi(n\,\triangle t\,, (j-1)\,\triangle x\,) \\
\triangle x &= (J-1)^{-1} \\
\triangle t &= \lambda\,\triangle x \qquad \lambda \equiv \text{“Courant number”}
\end{aligned}
$$

- We note in passing that the quantity λ defined above is often called the Courant number or Courant factor, after the great 20th century mathematician Richard Courant who was a pioneer in the study of finite difference solutions of time dependent PDEs (in particular, in the use of FD techniques to establish existence and uniqueness of such PDEs))

Uniform Grid for 1-D Wave Equation

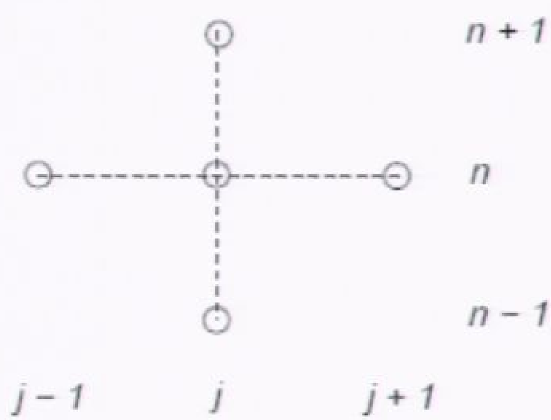


- When solving wave equations using FDAs, typically keep λ constant when $\triangle x$ varied.
- FDA will always be characterized by the *single* discretization scale, h.

$$\triangle x \equiv h$$

$$\triangle t = \lambda h$$

Stencil for "Standard" $O(h^2)$ Approximation of 1-D Wave Equation



FDA for 1-D Wave Equation

- Discretized Interior equation

$$
\begin{aligned}
(\triangle t)^{-2}\left(\phi_j^{n+1} - 2\phi_j^n + \phi_j^{n-1}\right) &= (\phi_{tt})_j^n + \frac{1}{12}\triangle t^2 (\phi_{tttt})_j^n + O(\triangle t^4) \\
&= (\phi_{tt})_j^n + O(h^2) \\
(\triangle x)^{-2}\left(\phi_{j+1}^{n} - 2\phi_j^n + \phi_{j-1}^{n}\right) &= (\phi_{xx})_j^n + \frac{1}{12}\triangle x^2 (\phi_{xxxx})_j^n + O(\triangle x^4) \\
&= (\phi_{xx})_j^n + O(h^2)
\end{aligned}
$$

Putting these two together, get $O(h^2)$ approximation

$$
\frac{\phi_j^{n+1} - 2\phi_j^n + \phi_j^{n-1}}{\triangle t^2} = \frac{\phi_{j+1}^{n} - 2\phi_j^n + \phi_{j-1}^{n}}{\triangle x^2} \qquad j = 2, 3, \cdots, J-1 \qquad (16)
$$

- Scheme such as (16) often called a *three level scheme* since couples *three "time levels"* of data (i.e. unknowns at three distinct, discrete times t^{n-1}, t^n, t^{n+1}.

FDA for 1-D Wave Equation

- Discretized Boundary conditions

$$\phi_1^{n+1} = \phi_J^{n+1} = 0$$

- Discretized Initial conditions
  - Need to specify *two* "time levels" of data (effectively $\phi(x,0)$ and $\phi_t(x,0)$), i.e. we must specify

$$\begin{aligned} \phi_j^0 \quad &, \quad j = 1, 2, \cdots, J \\ \phi_j^1 \quad &, \quad j = 1, 2, \cdots, J \end{aligned}$$

ensuring that the initial values are compatible with the boundary conditions.

- Can solve (16) *explicitly* for ϕ_j^{n+1}:

$$\phi_j^{n+1} = 2\phi_j^n - \phi_j^{n-1} + \lambda^2 \left(\phi_{j+1}^n - 2\phi_j^n + \phi_j^{n-1} \right) \quad (17)$$

FDA for 1-D Wave Equation

- Also note that (17) is actually a *linear system* for the unknowns $\phi_j^{n+1},\ j = 1, 2, \cdots, J$; in combination with the discrete boundary conditions can write

$$\mathbf{A}\ \phi^{n+1} = \mathbf{b} \tag{18}$$

  where $\mathbf{A}$ is a *diagonal* $J \times J$ matrix and ϕ^{n+1} and $\mathbf{b}$ are vectors of length J.

- Such a difference scheme for an IVP is called an *explicit* scheme.

3. Solution of Discrete Equations

Will not discuss in any detail until later this week

1-D Wave Equation: 1st Order Form

- Let us again consider the 1-D wave equation, solved on the spatial domain $0 \le x \le 1$, and where we will delay the specification of the boundary conditions for the time being

- We have

$$\phi_{tt} = \phi_{xx}, \quad 0 \le x \le 1, \quad t \ge 0 \tag{19}$$

$$\phi(0,x) = \phi_0(x) \tag{20}$$

$$\phi_t(0,x) = \Pi_0(x) \tag{21}$$

- We rewrite (19) in a form that involves only first time derivatives by defining the following auxiliary variables

$$\Phi(t,x) \equiv \phi_x \tag{22}$$

$$\Pi(t,x) \equiv \phi_t \tag{23}$$

FDA for 1-D Wave Equation

- Also note that (17) is actually a *linear system* for the unknowns $\phi_j^{n+1},\ j = 1, 2, \cdots, J$; in combination with the discrete boundary conditions can write

$$\mathbf{A}\ \phi^{n+1} = \mathbf{b} \tag{18}$$

where $\mathbf{A}$ is a *diagonal* $J \times J$ matrix and ϕ^{n+1} and $\mathbf{b}$ are vectors of length J.

- Such a difference scheme for an IVP is called an *explicit* scheme.

FDA for 1-D Wave Equation

- Discretized Boundary conditions

$$\phi_1^{n+1} = \phi_J^{n+1} = 0$$

- Discretized Initial conditions
  - Need to specify *two* "time levels" of data (effectively $\phi(x,0)$ and $\phi_t(x,0)$), i.e. we must specify

$$\begin{array}{ccc} \phi_j^0 & , & j = 1, 2, \cdots, J \\ \phi_j^1 & , & j = 1, 2, \cdots, J \end{array}$$

  ensuring that the initial values are compatible with the boundary conditions.

- Can solve (16) *explicitly* for ϕ_j^{n+1}:

$$\phi_j^{n+1} = 2\phi_j^n - \phi_j^{n-1} + \lambda^2 \left(\phi_{j+1}^n - 2\phi_j^n + \phi_j^{n-1} \right) \tag{17}$$

FDA for 1-D Wave Equation

- Also note that (17) is actually a *linear system* for the unknowns $\phi_j^{n+1},\ j = 1, 2, \cdots, J$; in combination with the discrete boundary conditions can write

$$\mathbf{A}\ \phi^{n+1} = \mathbf{b} \tag{18}$$

where $\mathbf{A}$ is a *diagonal* $J \times J$ matrix and ϕ^{n+1} and $\mathbf{b}$ are vectors of length J.

- Such a difference scheme for an IVP is called an *explicit* scheme.

FDA for 1-D Wave Equation

- Discretized Boundary conditions

$$\phi_1^{n+1} = \phi_J^{n+1} = 0$$

- Discretized Initial conditions
  - Need to specify *two* "time levels" of data (effectively $\phi(x, 0)$ and $\phi_t(x, 0)$), i.e. we must specify

$$\phi_j^0 \quad , \quad j = 1, 2, \cdots, J$$
$$\phi_j^1 \quad , \quad j = 1, 2, \cdots, J$$

  ensuring that the initial values are compatible with the boundary conditions.

- Can solve (16) *explicitly* for ϕ_j^{n+1}:

$$\phi_j^{n+1} = 2\phi_j^n - \phi_j^{n-1} + \lambda^2 \left(\phi_{j+1}^n - 2\phi_j^n + \phi_j^{n-1} \right) \qquad (17)$$

FDA for 1-D Wave Equation

- Discretized Boundary conditions

$$\phi_1^{n+1} = \phi_J^{n+1} = 0$$

- Discretized Initial conditions
  - Need to specify *two* "time levels" of data (effectively $\phi(x,0)$ and $\phi_t(x,0)$), i.e. we must specify

$$\phi_j^0 \quad , \quad j = 1, 2, \cdots, J$$
$$\phi_j^1 \quad , \quad j = 1, 2, \cdots, J$$

ensuring that the initial values are compatible with the boundary conditions.

- Can solve (16) *explicitly* for ϕ_j^{n+1}:

$$\phi_j^{n+1} = 2\phi_j^n - \phi_j^{n-1} + \lambda^2 \left(\phi_{j+1}^n - 2\phi_j^n + \phi_j^{n-1} \right) \quad (17)$$

FDA for 1-D Wave Equation

- Also note that (17) is actually a *linear system* for the unknowns $\phi_j^{n+1},\ j = 1, 2, \cdots, J$; in combination with the discrete boundary conditions can write

$$\mathbf{A}\ \phi^{n+1} = \mathbf{b} \tag{18}$$

where $\mathbf{A}$ is a *diagonal* $J \times J$ matrix and ϕ^{n+1} and $\mathbf{b}$ are vectors of length J.

- Such a difference scheme for an IVP is called an *explicit* scheme.

FDA for 1-D Wave Equation

- Also note that (17) is actually a *linear system* for the unknowns $\phi_j^{n+1},\ j = 1, 2, \cdots, J$; in combination with the discrete boundary conditions can write

$$\mathbf{A}\ \phi^{n+1} = \mathbf{b} \tag{18}$$

where $\mathbf{A}$ is a *diagonal* $J \times J$ matrix and ϕ^{n+1} and $\mathbf{b}$ are vectors of length J.

- Such a difference scheme for an IVP is called an *explicit* scheme.

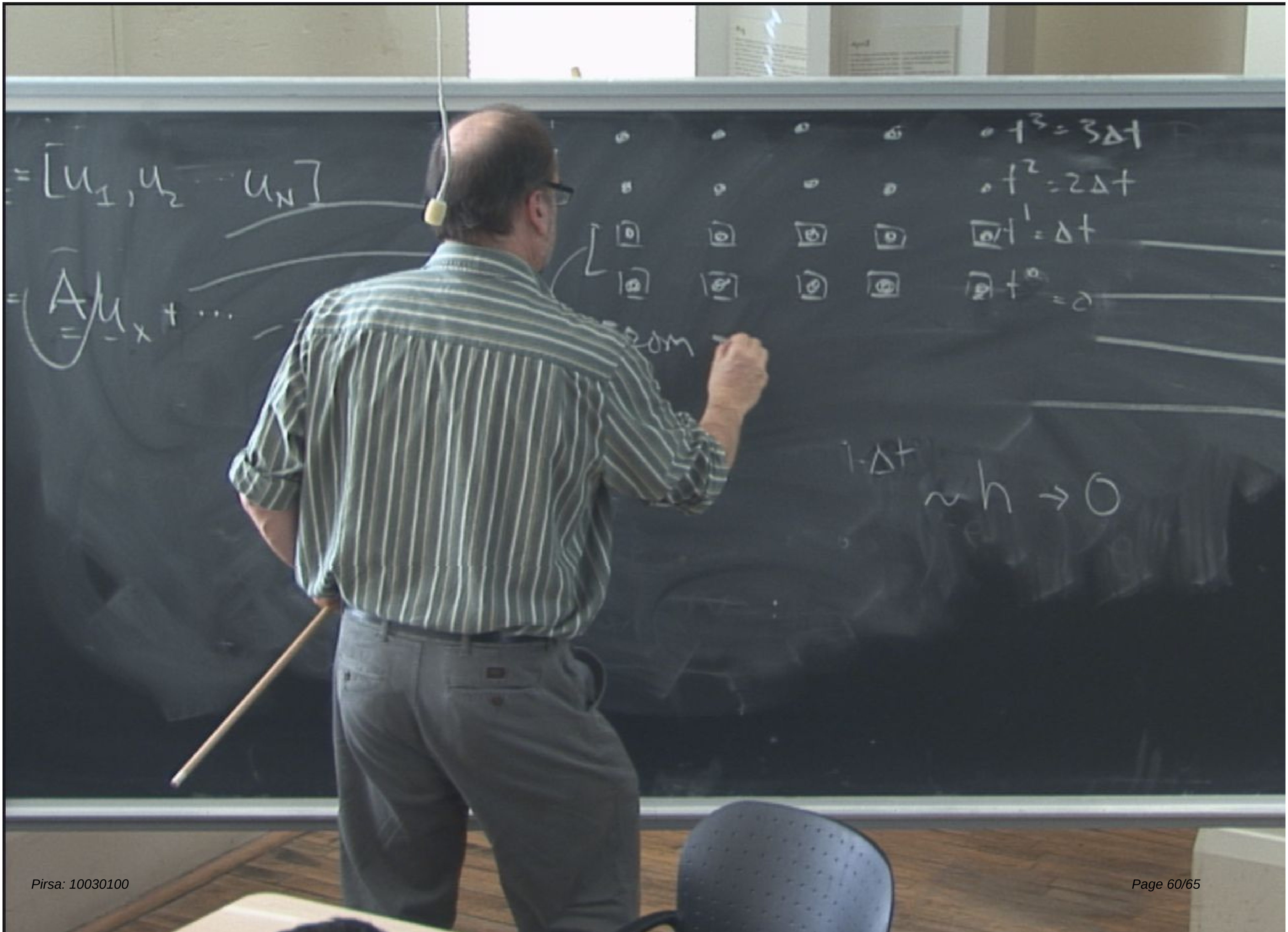
FDA for 1-D Wave Equation

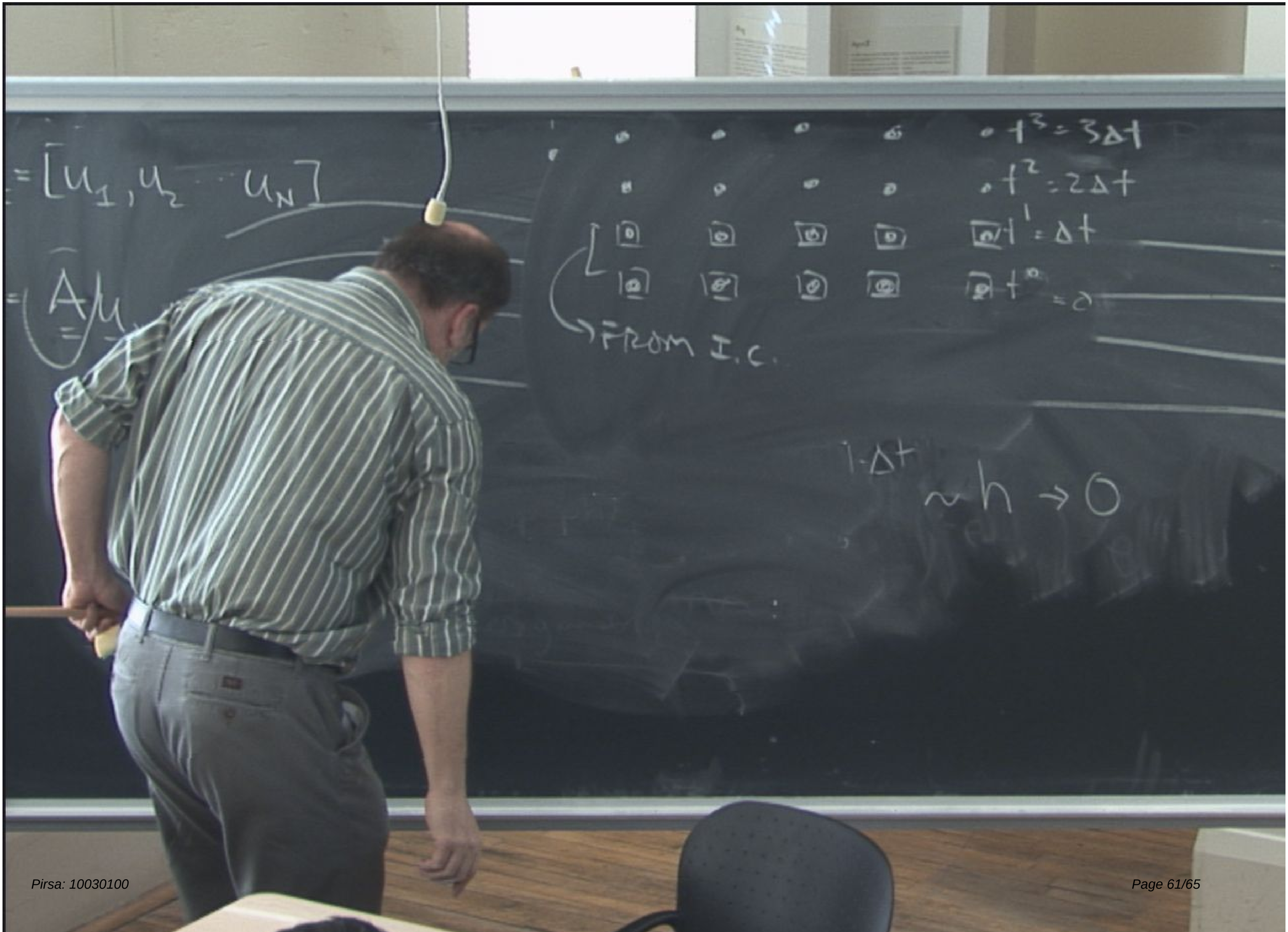
- Also note that (17) is actually a *linear system* for the unknowns $\phi_j^{n+1},\ j = 1, 2, \cdots, J$; in combination with the discrete boundary conditions can write

$$\mathbf{A}\ \phi^{n+1} = \mathbf{b} \tag{18}$$

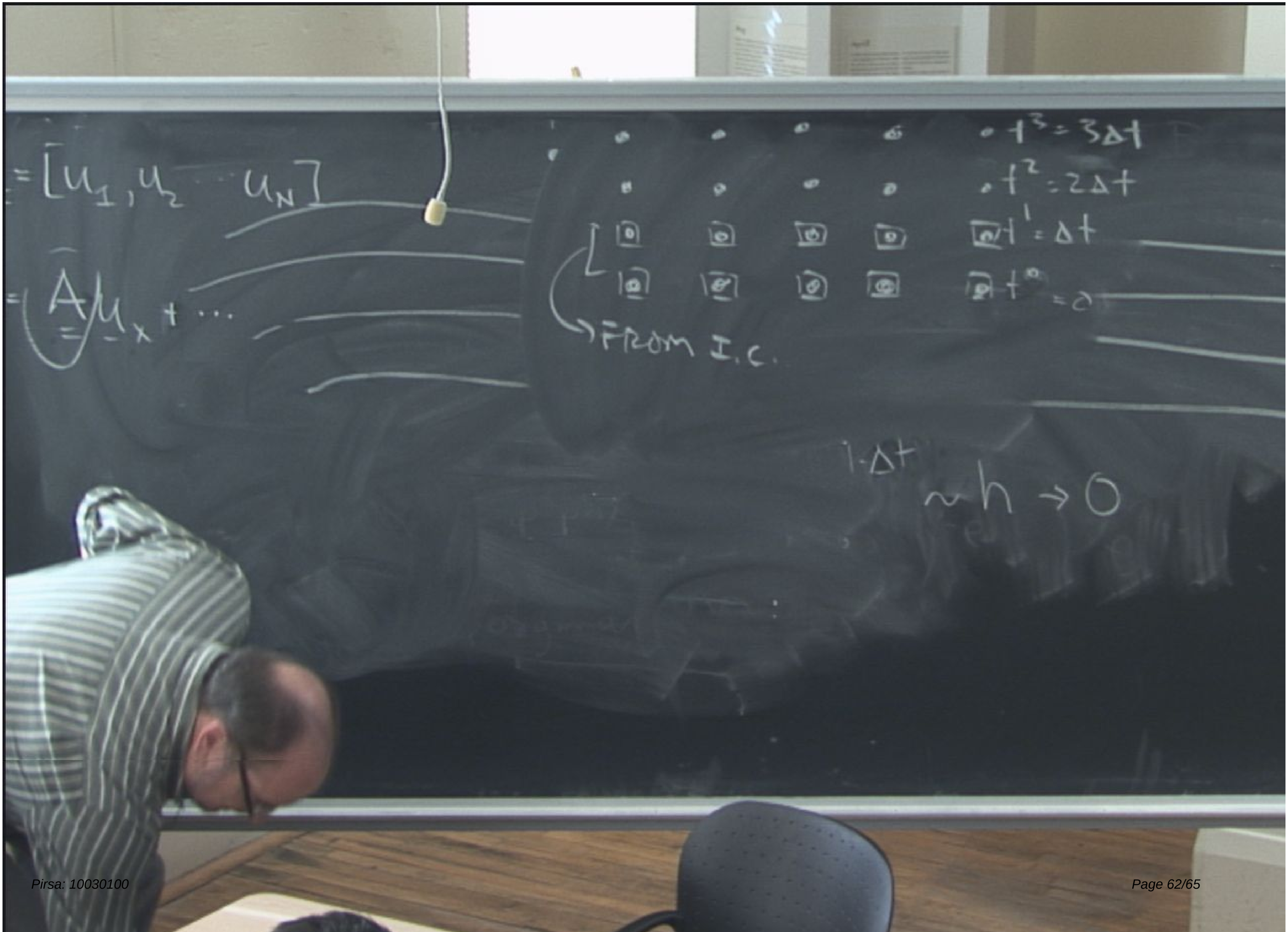
where $\mathbf{A}$ is a *diagonal* $J \times J$ matrix and ϕ^{n+1} and $\mathbf{b}$ are vectors of length J.

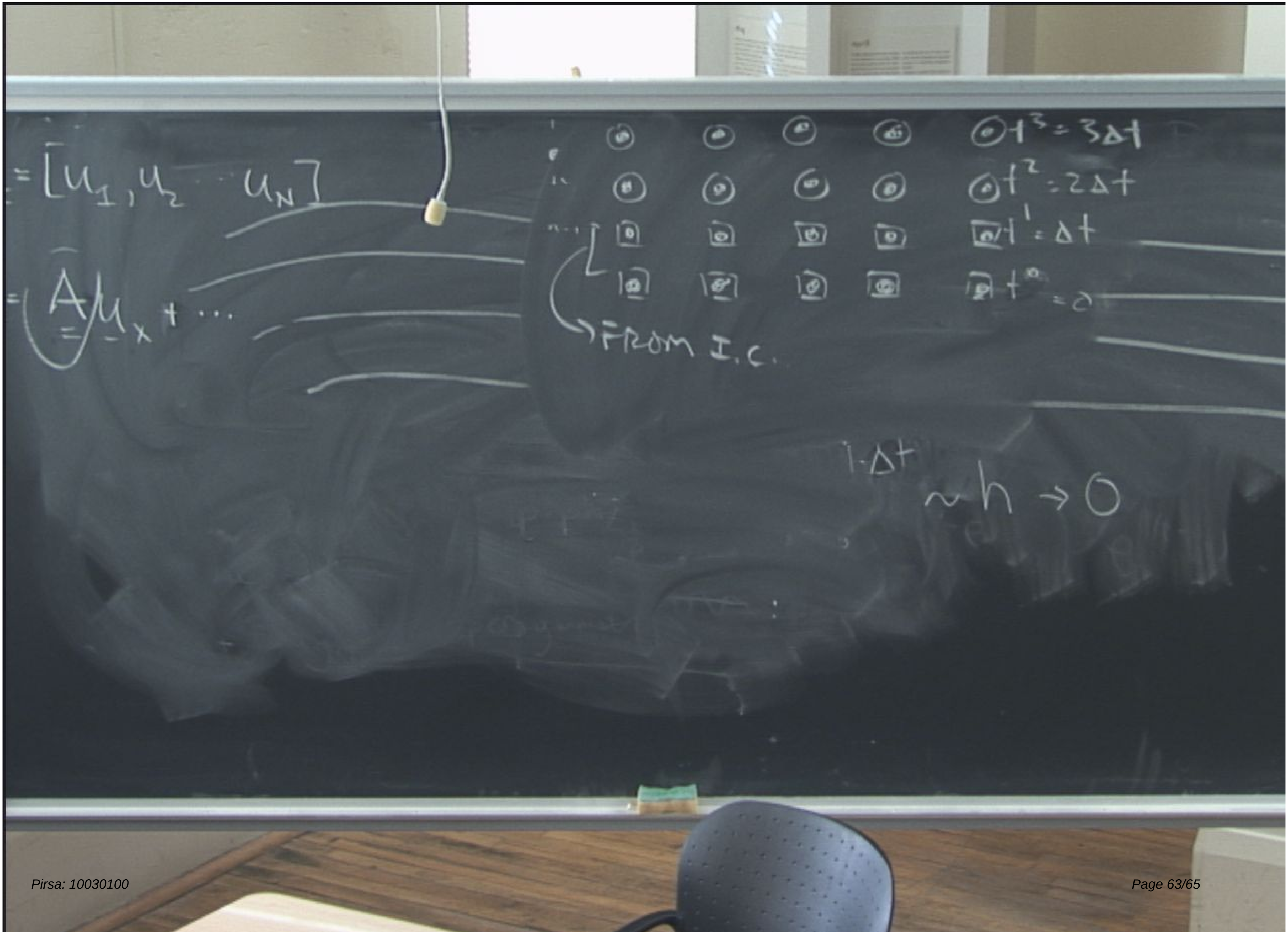
- Such a difference scheme for an IVP is called an *explicit* scheme.


= [u1, u2 ... uN]
= A u_x + ...
t^3 = 3Δt
t^2 = 2Δt
t^1 = Δt
t^0 = 0
1-Δt
~ h → 0



$=[u_1, u_2 \cdots u_N]$
$t^3 = 3\Delta t$
$t^2 = 2\Delta t$
$t^1 = \Delta t$
$t^0 = 0$
FROM I.C.
$1 \cdot \Delta t$
$\sim h \to 0$


= [u_1, u_2 ··· u_N]
= A u_x + ···
t^3 = 3Δt
t^2 = 2Δt
t^1 = Δt
t^0 = 0
FROM I.C.
1·Δt
~ h → 0


= [u1, u2 ... uN]
= A ux + ...
t3 = 3Δt
t2 = 2Δt
t1 = Δt
t0 = 0
FROM I.C.
1·Δt
~ h → 0

FDA for 1-D Wave Equation

- Discretized Interior equation

$$
\begin{aligned}
(\triangle t)^{-2}\left(\phi_j^{n+1} - 2\phi_j^n + \phi_j^{n-1}\right) &= (\phi_{tt})_j^n + \frac{1}{12}\triangle t^2 (\phi_{tttt})_j^n + O(\triangle t^4) \\
&= (\phi_{tt})_j^n + O(h^2) \\
(\triangle x)^{-2}\left(\phi_{j+1}^{n} - 2\phi_j^n + \phi_{j-1}^{n}\right) &= (\phi_{xx})_j^n + \frac{1}{12}\triangle x^2 (\phi_{xxxx})_j^n + O(\triangle x^4) \\
&= (\phi_{xx})_j^n + O(h^2)
\end{aligned}
$$

Putting these two together, get $O(h^2)$ approximation

$$
\frac{\phi_j^{n+1} - 2\phi_j^n + \phi_j^{n-1}}{\triangle t^2} = \frac{\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n}{\triangle x^2} \qquad j = 2, 3, \cdots, J-1 \qquad (16)
$$

- Scheme such as (16) often called a *three level scheme* since couples *three "time levels"* of data (i.e. unknowns at three distinct, discrete times t^{n-1}, t^n, t^{n+1}.

Stencil for "Standard" $O(h^2)$ Approximation of 1-D Wave Equation