BOUNDARY CONDITIONS

1

A. Ponev

CFD Spring 2013

Consider adv-diff equation with BC:

$$u_t + a u_x = d u_{xx}, \quad t > 0$$

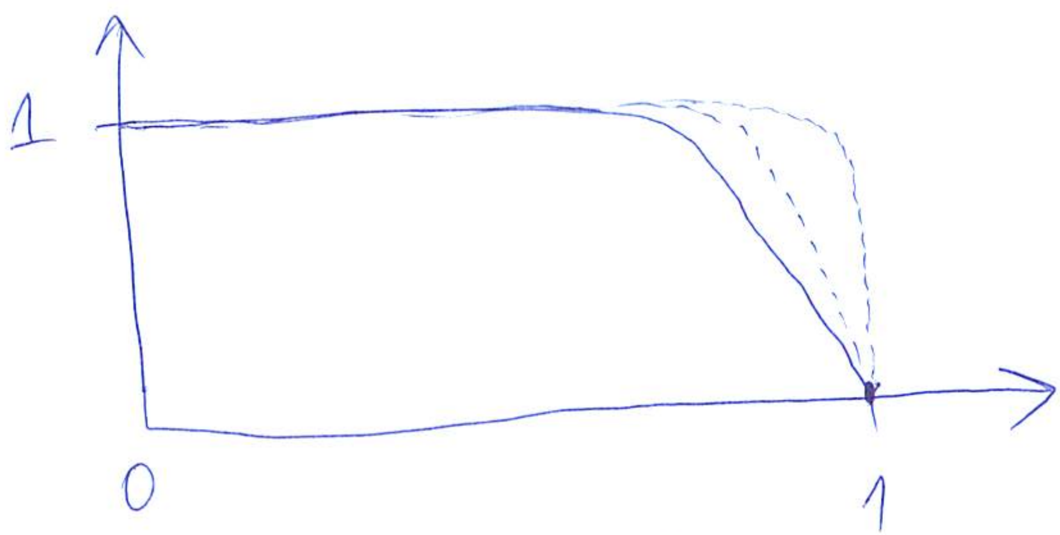
$$0 < x < L = 1$$

$$\begin{cases} u(0,t) = 1 \\ u(1,t) = 0 \quad \text{Dirichlet} \end{cases}$$

or $u_x(1,t) = 0$ Neumann

For Dirichlet, the steady-state solution $u_t = 0 \Rightarrow a u_x = d u_{xx}$

$$u(x,t) = \left[e^{a/d} - e^{ax/d}\right] / \left[e^{a/d} - 1\right]$$



Boundary layer of width $\sim d/a$

The solution forgets the boundary condition after a distance $\Delta x \sim d/a$ away from the boundary. If

$$P_e = \left(\frac{\Delta x}{L}\right)^{-1} = \frac{aL}{d} \gg 1$$

this layer will be **very thin**!
(compare to AIRPLANE wing)

If we used Neumann BC, ③
however, we would get

$$u(x,t) = 1$$

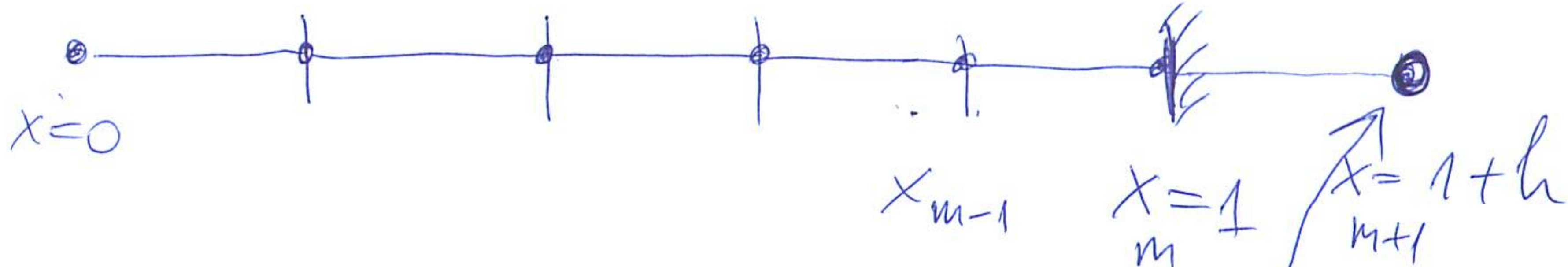
with <u>no</u> boundary layer.

This is what we often do in CFD if it is not important to resolve the boundary layer: We make it disappear by using a <u>fake</u> boundary condition (extrapolation from interior), as we will see next.

For now, assume <u>smooth</u> solution!

Consider first advection-dominated flow:

$$\begin{cases} u_t + u_x = 0, & x \in (0,1) \\ u(0,t) = g_0(t) & \text{inflow} \\ \text{no BC at outflow boundary} \end{cases}$$



Every CFD code has them!

"ghost" cell or "virtual" cell

Assume we use centered difference in the interior

5

$$w_j' = \frac{1}{2h}\left(w_{j-1} - w_{j+1}\right)$$

At the <u>left boundary</u>,

$w_0(t) = g_0(t)$ follows from BC

At the <u>right boundary</u>, naive application of the same stencil leads to

$$w_m' = \frac{1}{2h}\left(w_{m-1} - w_{m+1}\right)$$

But w_{m+1} is <u>not</u> a real variable.

Since there is no BC at right, the best we can do is <u>extrapolate</u> from the interior

$w_{m+1} = w_m$ — constant extrapolation

which leads to $w_m' = \frac{1}{2h}(w_{m-1} - w_m)$

or <u>linear extrapolation</u>

$$w_{m+1} = 2w_m - w_{m-1} \Rightarrow$$

$$w_m' = \frac{1}{2h}\left[w_{m-1} - 2w_m + w_{m-1}\right]$$

$$w_m' = \frac{1}{h}(w_{m-1} - w_m)$$

which means we are using the <u>upwind scheme</u> at the right boundary.

This makes sense!

So let us choose

$$\boxed{w_m' = \frac{1}{h}\left(w_{m-1} - w_m\right)}$$

which we can either implement on the code as a one-sided special stencil at the boundary, or, we can keep the same stencil but use a ghost cell $w_{m+1} = 2w_m - w_{m-1}$ (better?)

The problem we have now is that we switched from a second order to first order local spatial truncation error near the boundary.

We can prove that the scheme is <u>still second-order accurate</u>! 8

Denote the local spatial truncation error with σ_h as before, and consider

$$w' = A\,w + g(t)$$

and <u>assume</u> that

$$\boxed{\sigma_h = A\,\xi(t) + \eta(t)}$$

Then, the global error

$$\varepsilon' = A\,\varepsilon + \sigma_h = A(\varepsilon + \xi) + \eta$$

Take norms after variation-of-constants, exactly as before, to get

(9)

$$\tilde{\varepsilon} = \varepsilon + \xi$$

$$\tilde{\varepsilon}' = A\tilde{\varepsilon} + \xi'(t) + \eta(t)$$

$$\boxed{\|\varepsilon(t)\| \le \|\xi\| + K e^{\omega t} \|\varepsilon(0) + \xi(0)\| + \frac{K}{\omega}(e^{\omega t} - 1) \max_{0 \le s \le t} \|\xi'(s) + \eta(s)\|}$$

This means that

<u>Theorem</u>: If $\boxed{\|\xi\|, \|\xi'\|, \|\eta\| \le C h^r}$ $\forall t$

and $\|\varepsilon(0)\| \le C_0 h^r$, then

$$\boxed{\|\varepsilon(t)\| \le \tilde{C}(t) h^r}$$ $\Leftarrow$ global order $=$ local order

the difference with before is that (10)
this requires a bound on

$\|\xi(t)\|$ and $\|\xi'(t)\|$ and $\|\eta(t)\|$

NOT on $\|\sigma_h(t)\|$. Recall

$$\sigma_h = A\xi + \eta \Rightarrow \xi \simeq A^{-1}\sigma_h$$

Note that ξ_h is the solution of the steady-state problem with forcing $\boxed{g = \eta - \sigma_h}$. So the local error acts as a source term that then propagates through the global solution, spatially.

Let's now go back to our example:

$$A = \frac{1}{2h} \begin{bmatrix} 0 & -1 & & & \\ 1 & \ddots & -1 & & \\ & 1 & \ddots & \ddots & \\ & & \ddots & \ddots & \\ & & 1 & 0 & -1 \\ & & & 2 & -2 \end{bmatrix} \leftarrow \text{boundary}$$

$$\begin{cases} \sigma_{h,m} = u_t(x_m,t) - \frac{1}{h}\left[u(x_{m-1},t) - u(x_m,t)\right] \\ \qquad = -\frac{1}{2} h\, u_{xx}(x_m,t) + O(h^2) \end{cases}$$

Assume smoothness $\|u_{xx}\| \leq c_1$

$$\Rightarrow \quad \sigma_{h,m} \leq c_1 h$$

$$\sigma_{h,j<m} \leq c_2 h^2$$

So we have

$$\sigma_h = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 1 \end{bmatrix} c_1 h + \underbrace{O(h^2)}_{\eta}$$

and we need to solve

$$\sigma_h = A\xi + \eta \Rightarrow$$

$$\frac{1}{2h}\begin{bmatrix} 0 & -1 & & \\ 1 & \ddots & \ddots & \\ & \ddots & 0 & \\ & & 2 & -2 \end{bmatrix} \xi = \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix} c_1 h$$

This system we can solve by hand

$$\xi = [\ \ldots, \xi_m, 0, \xi_m, 0, \xi_m]$$

where $\xi'_m = C_1 h^2$ note we gained! one power

So indeed $\|\xi'\| = O(h^2)$, $\|\eta\| = O(h^2)$

and the theorem tells us the spatial semi-discretization is <u>second-order accurate</u>!

This is a very common occurence, and we often do low-order discretizations near boundaries, even <u>zeroth-order</u> (!?) (inconsistent) stencils next to boundaries.

the hard part is $\boxed{\text{stability}}$

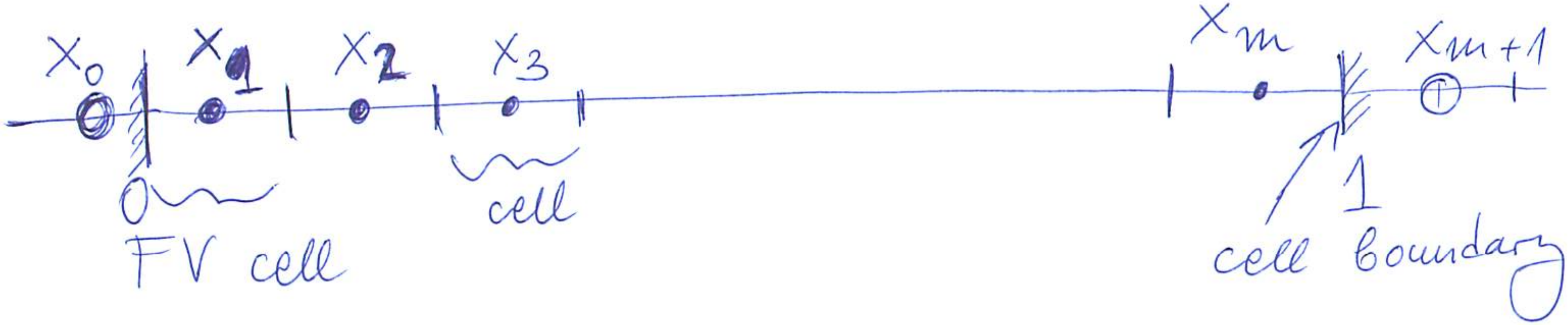
there is no simple general way to analyze stability, i.e., to show that $\boxed{\| e^{tA} \| \leq K e^{\omega t}}$ uniformly in h (14)

The wisdom is that for schemes with some (artificial or real) dissipation

sufficient? stability is ensured by damping of any instabilities near the boundaries.

Often we rely on numerical experiments since instabilities are typically obvious.

For finite-volume schemes often boundaries overlap with the faces of the grid, i.e, half-integer points in 1D:



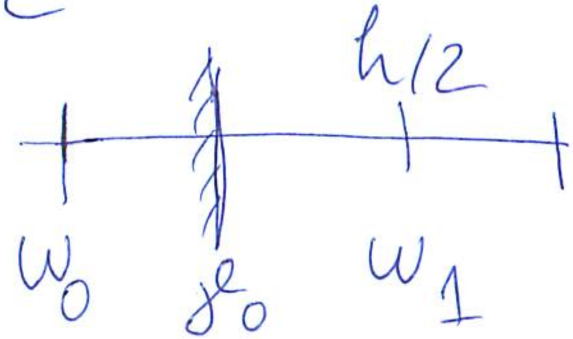
$$x_j = \left(j - \tfrac{1}{2}\right) h \quad , \quad h = 1/m$$

$$u_t = u_{xx} \quad \text{with BC} \quad \begin{cases} u(0,t) = g_0(t) \\ u_x(1,t) = g_1(t) \end{cases}$$

We need ghost values w_0 and w_{m+1}

Left: $\frac{w_0 + w_1}{2} = \gamma_0$ ← linear ext/interpolation of BC

$\Rightarrow \boxed{w_0 = 2\gamma_0 - w_1}$



Right: What Neumann BC really means is no-flux through $x = 1$:

$$F(x=1) = \frac{w_{m+1} - w_m}{\Delta x} = 0 \Rightarrow$$

$\boxed{w_{m+1} = w_m}$ (constant extrapolation)

Now consider advection - diffusion

with finite - volume

$$u_t + a u_x = d u_{xx}$$

Dirichlet BC gives flux directly

for advection, $f_{adv} = a u$

Neumann BC gives flux directly

for diffusion, $f_{diff} = -d u_x$

So for those fluxes we do not

need any interpolation / extrapolation

or ghost cells!

If we want our method to work for $d \to 0$ (pure advection or <u>advection-dominated</u> flows, we

2*

<u>must</u> treat advection separately and realize that there is <u>no BC at outflow</u> for advection, only for diffusion.

So treat diffusion as already explained in lecture notes

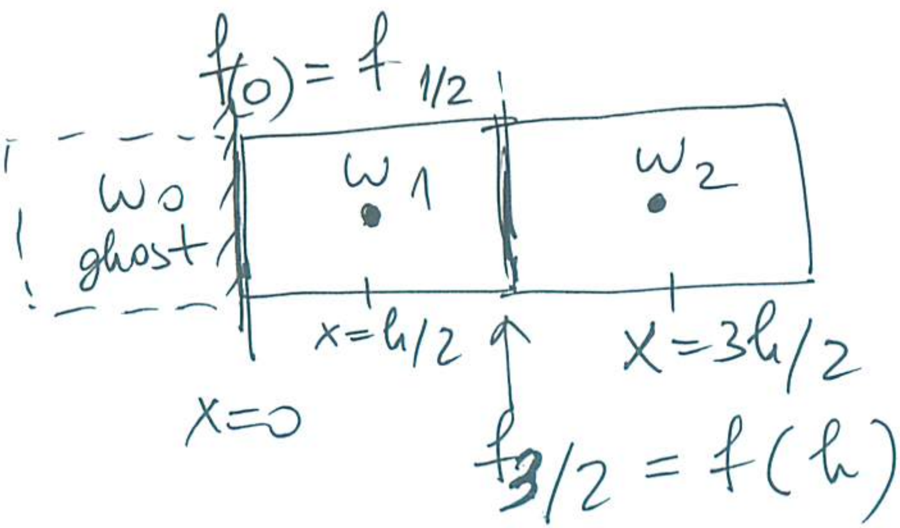
For advection, we need to consider second or third-order schemes separately.

3*

$$\begin{cases} u(0,t) = \tilde{f}(t) & \text{Dirichlet on left} \\ u_x(L_x, t) = \tilde{g}(t) & \text{for diffusion but} \end{cases}$$

does **not matter** for advection!

At the right, we have outflow boundary and at the left, inflow boundary



$$f_{1/2} = f(0) = a\,\tilde{f}(t)$$

(flux at $x=0$)

$$f_{3/2} \approx f(x=h) = ?$$

An obvious choice is

4*

(1) <u>Centered - advection:</u>

$$f_{3/2} = -a \, \frac{w_1 + w_2}{2} \quad \text{(no ghost cell)}$$

this gives (this is <u>crucial</u> to check)

$$\frac{dw_1}{dt} = -\frac{a}{h}\left(w_1 + w_2 - w_{1/2}\right)$$

where $\quad w_{1/2} \simeq u(x=0) = \tilde{f}(t)$

A Taylor series expansion shows that this is <u>first-order</u> accurate at the boundary, which is ok since we get +1 order from 1/h in stencil.

<u>NOTE</u>: One can <u>pretend</u> here this is finite-difference

At the outflow boundary, we want a one-sided stencil (no BC) so it is most natural to use first-order upwinding at outflow 5*

$$\frac{dw_N}{dt} = -\frac{a}{h}\left(w_N - w_{N-1}\right)$$

Simple algebra shows that this is the same as using a ghost cell with linear extrapolation (MUST CHECK THIS!)

$$w_{N+1} = 2w_N - w_{N-1}$$