Curved Boundary Conditions
in lattice Boltzmann method

Goncalo Silva
Department of Mechanical Engineering
Instituto Superior Técnico (IST)
Lisbon, Portugal

Outline

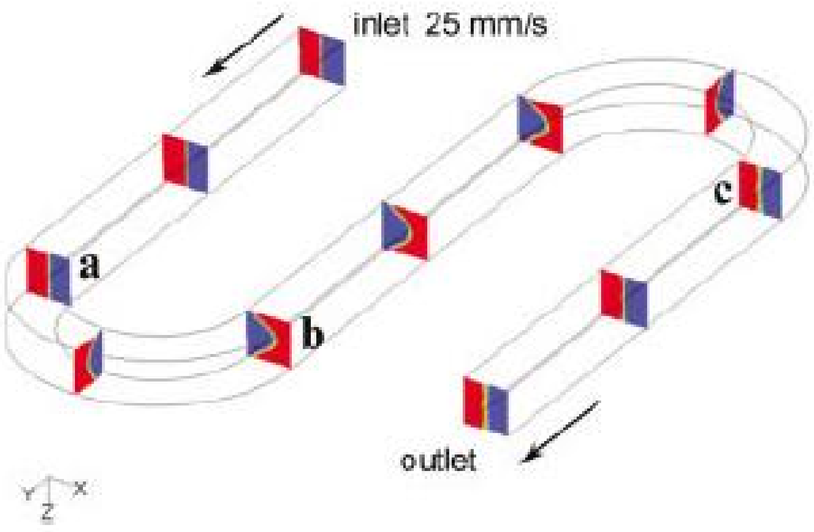
1. Introduction
   - General aspects
   - Motivation
   - Possible approaches
2. Problem statement
   - Off-grid populations
   - Interpolated Boundary Conditions
3. Summary

Outline

Irregular geometries in fluid flow problems

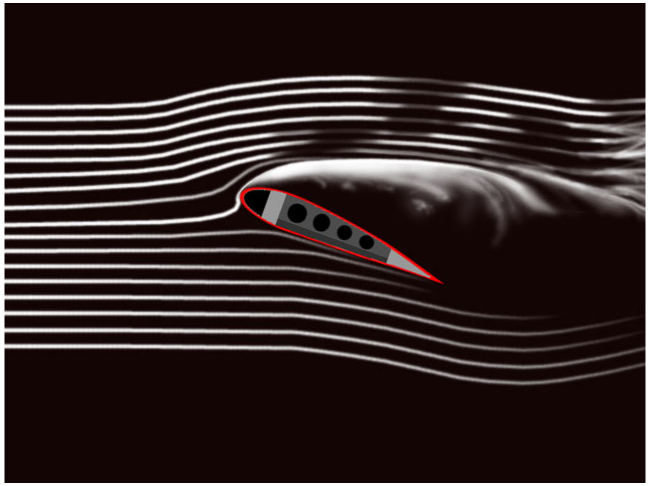
Most practical fluid flow problems of interest in engineering or science involve:

1) **irregular** domains



Yamaguchi *et al.* AIChE, 50(7), 1530 (2004)

2) **irregular** shaped objects

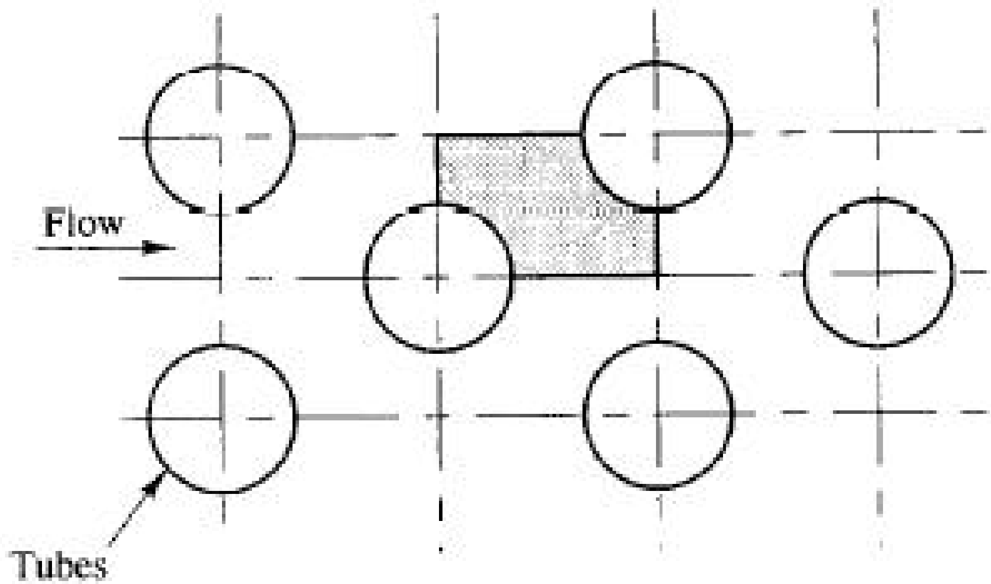


http://engineeringskills.wikidot.com/concepts#toc27

Computational Fluid Dynamics: non-orthogonal boundaries

Representing irregular domains in Computational Fluid Dynamics

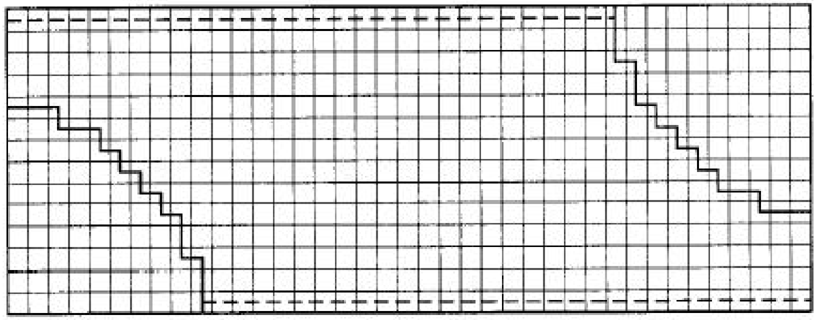
Example: Flow in a tube bank



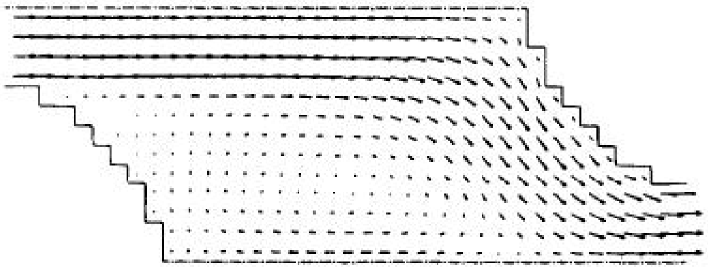
Versteeg & Malalasekera, An Introduction to CFD. Prentice Hall, 1995

Computational Fluid Dynamics: non-orthogonal boundaries

Cartesian grid

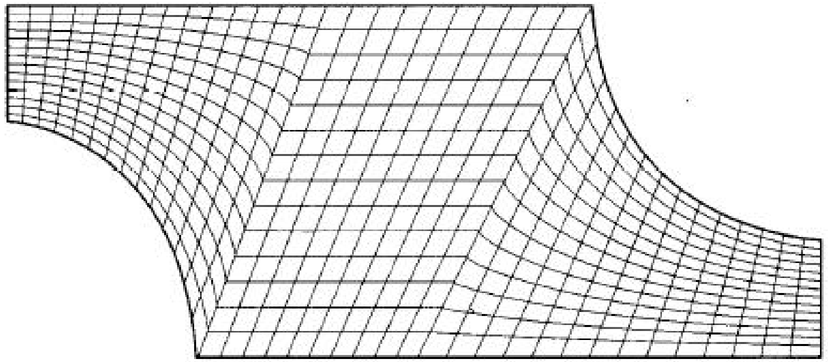


(40 × 15 grid)

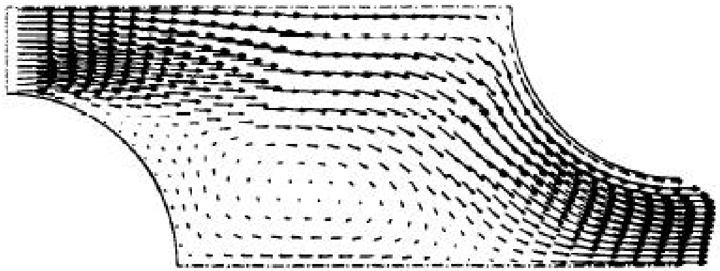


Versteeg & Malalasekera, An Introduction to CFD.
Prentice Hall, 1995

Non-orthogonal body-fitted grid



(40 × 15 grid)

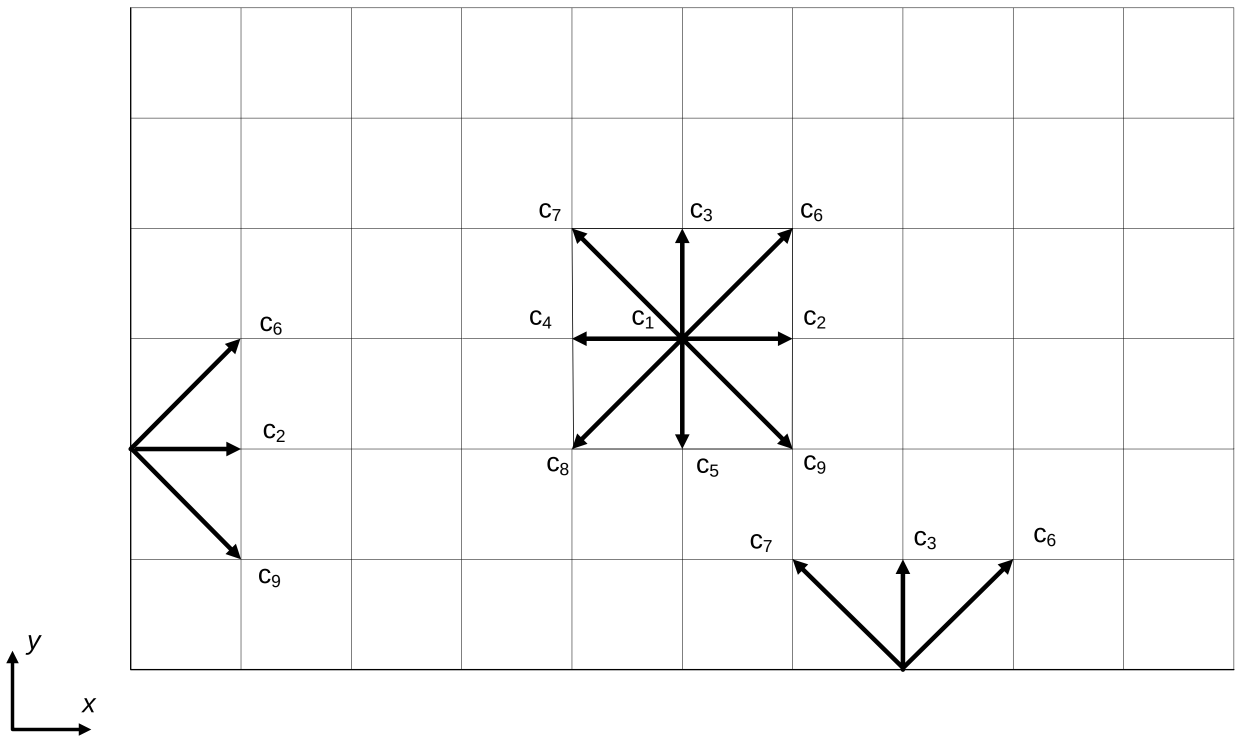


Versteeg & Malalasekera, An Introduction to CFD.
Prentice Hall, 1995

Lattice Boltzmann method: non-orthogonal boundaries

Lattice Boltzmann algorithm consists of...

...stream along links and equilibrate at nodes

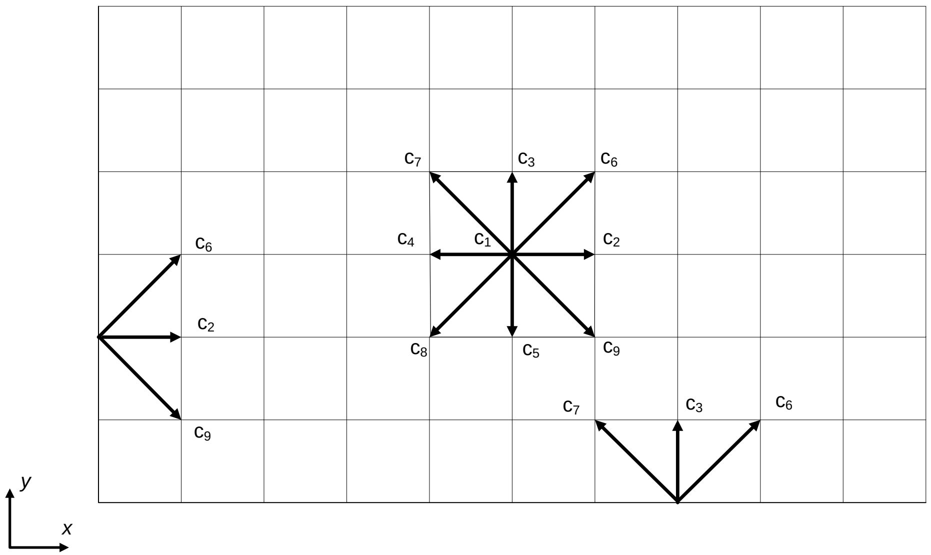


Lattice Boltzmann method: non-orthogonal boundaries

⇒ In "stream along links and equilibrate at nodes" philosophy the **streaming operation is exact**

→ **Advantage:** No interpolations are needed → virtually no numerical dissipation is introduced in streaming

→ **Disadvantage:** Lattice structure, *i.e.* velocity space discretization, constrains the configuration space discretization, *i.e.* the location of spatial nodes is prescribed by lattice

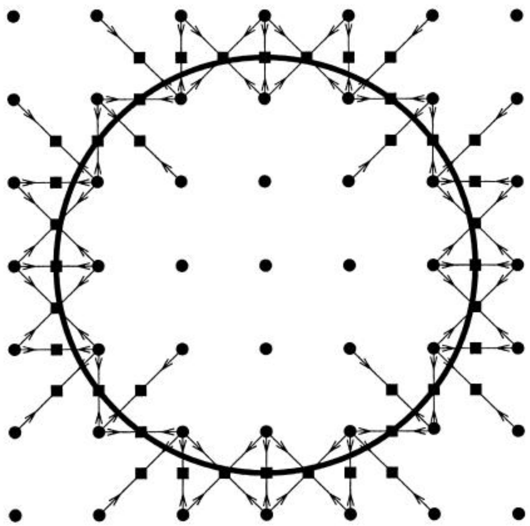


Lattice Boltzmann method: non-orthogonal boundaries

Conclusion:

Because the lattice Boltzmann method is an "on-grid" scheme it is restricted to **uniform cartesian grids**

Are we limited to the stepwise representation of irregular shapes?



Ladd & Verbeg 2001, J. Stat. Phys. 104, 1191-1251

Lattice Boltzmann method: non-orthogonal boundaries

The fact is...

...if we intend to represent non-orthogonal shapes we have to use numerical approximations (*e.g.* interpolations) somewhere

Two different approaches:

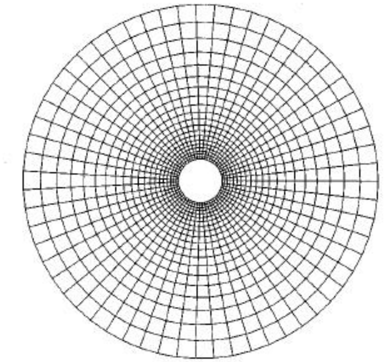
- Interpolate: Everywhere
- Preserve: Boundary scheme
- **Non-orthogonal body-fitted grids**

- Interpolate: Boundary scheme
- Preserve: Everywhere
- **Interpolated boundary conditions**

Lattice Boltzmann method: non-orthogonal boundaries

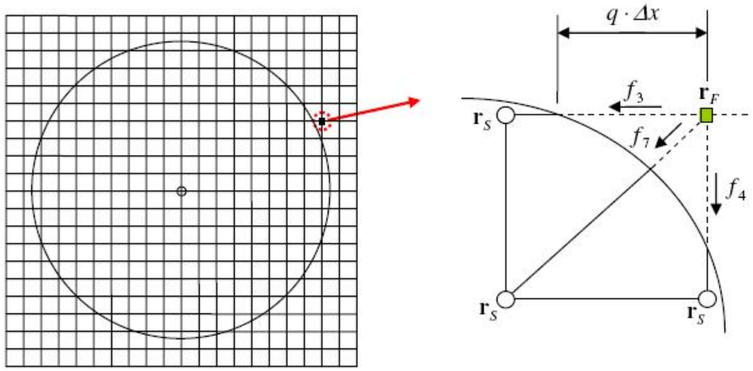
Example: Flow domain containing a solid circle represented through each approach:

- **Non-orthogonal body-fitted grids**



He & Doolen 1997, Phys. Rev. E 56, 434

- **Interpolated boundary conditions**

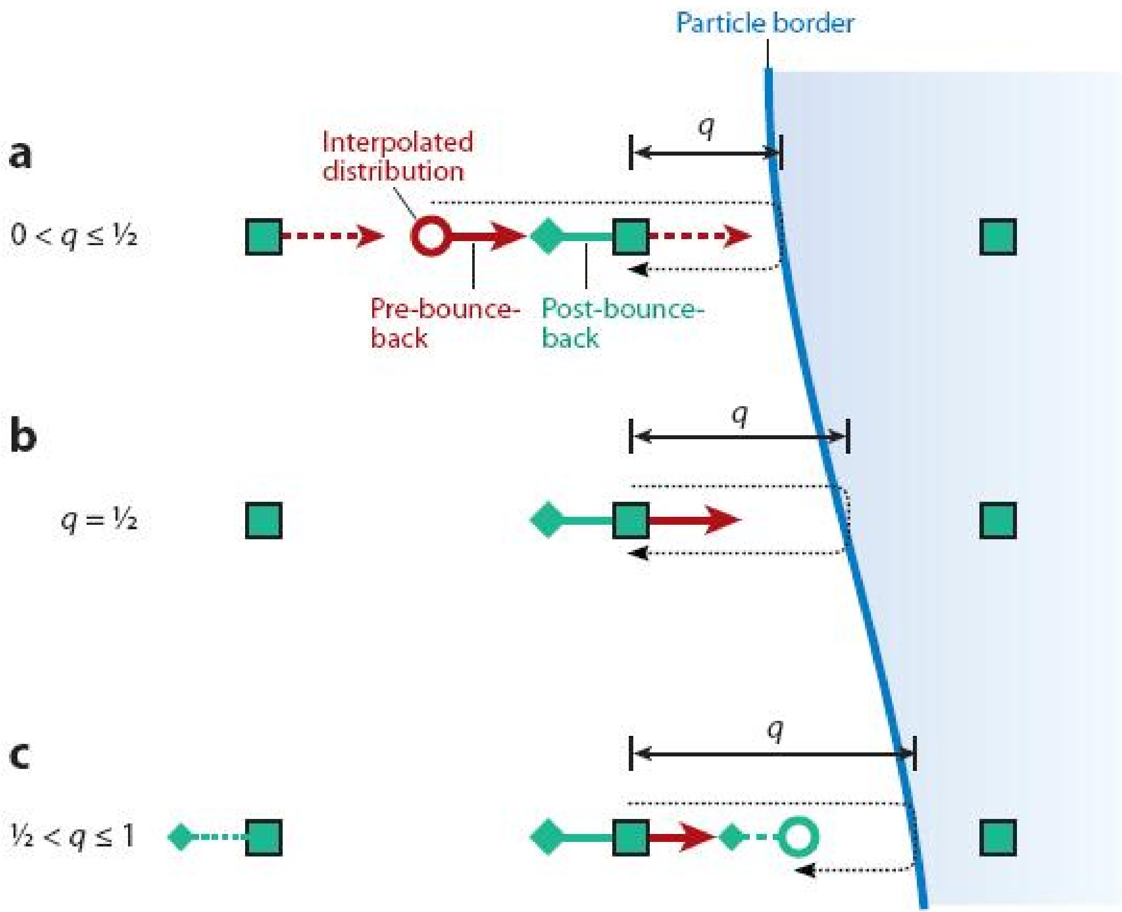


Kao & Yang 2008, J. Comp. Phys. 227, 5671-5690

Outline

Half-way bounceback

Example:

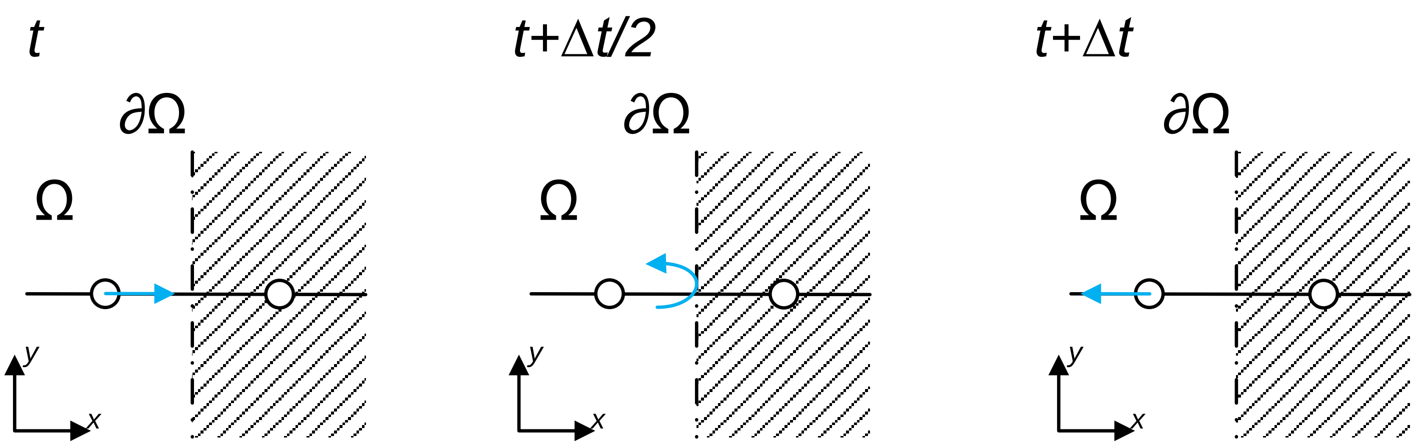


Aidun & Clausen 2010, Annu. Rev. Fluid Mech. 42, 439-472

Half-way bounceback

Remember: When the wall is half-way

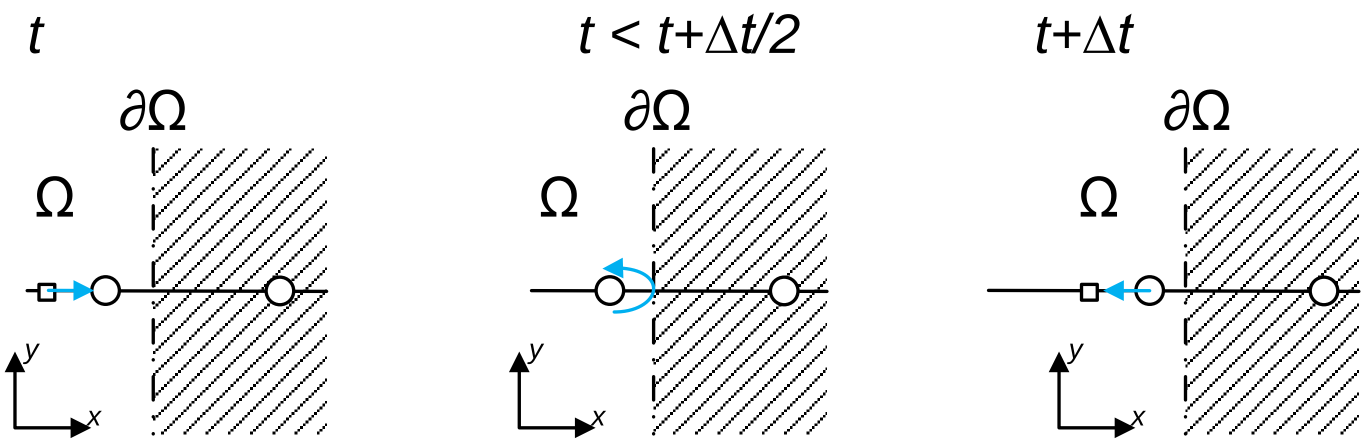
- **On-grid** population at time t is propagated
- Bounces-back at the wall
- Returns to the same **on-grid** location at time $t + \Delta t$



Half-way bounceback

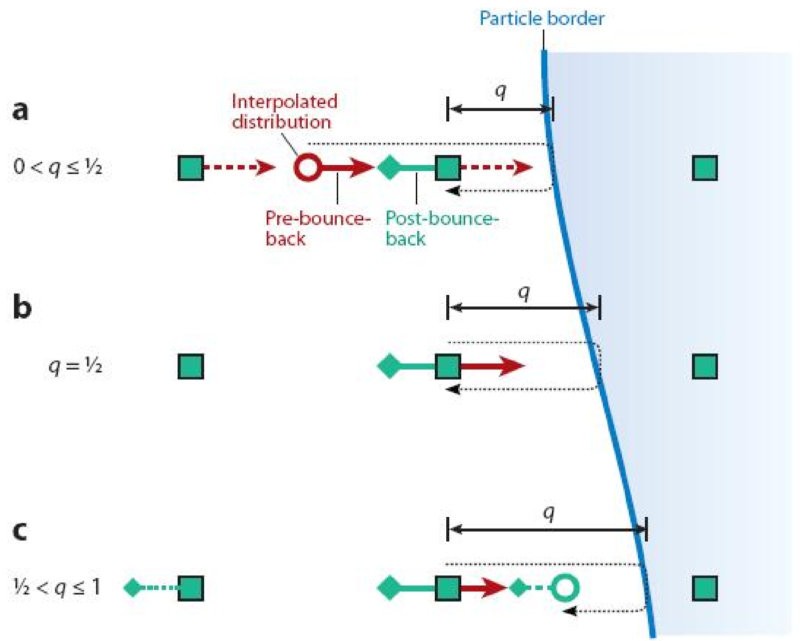
However: When the wall is **NOT** half-way

- Construct **off-grid** population at time t from where is propagated
- Bounces-back at the wall
- Returns to an **on-grid** location at time $t + \Delta t$



Half-way bounceback

Coming back to the example:



Aidun & Clausen 2010, Annu. Rev. Fluid Mech. 42, 439-472

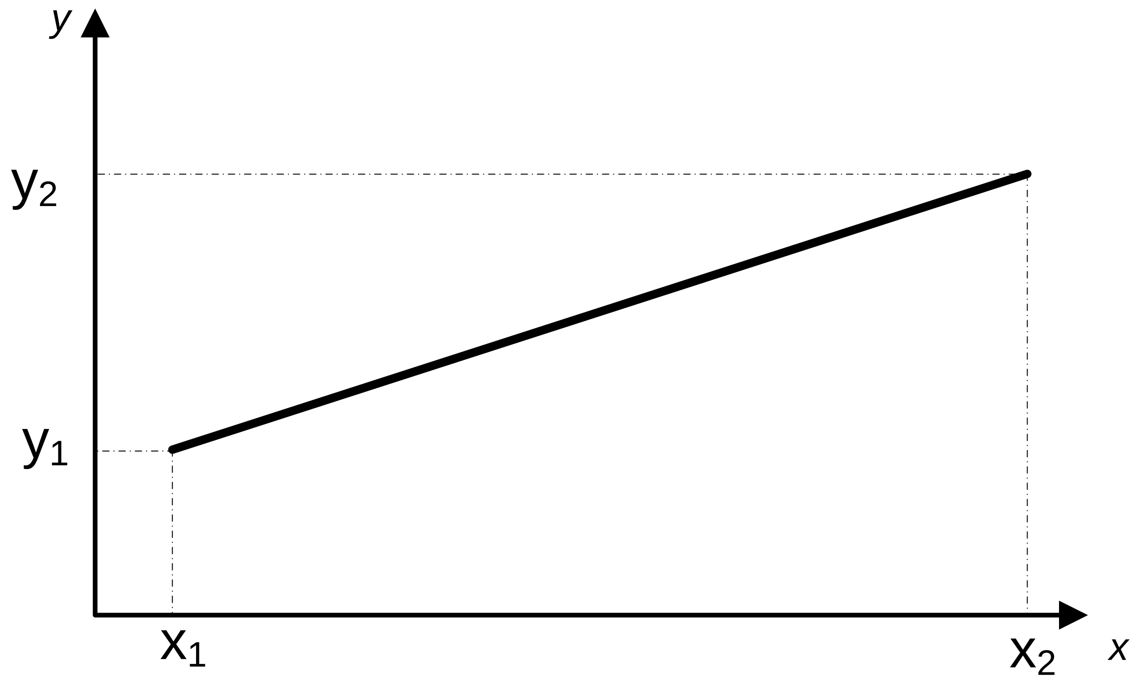
Idea of interpolating the boundary condition:

Reconstruct the **off-grid** population by interpolating the known **on-grid** data

Linear interpolation

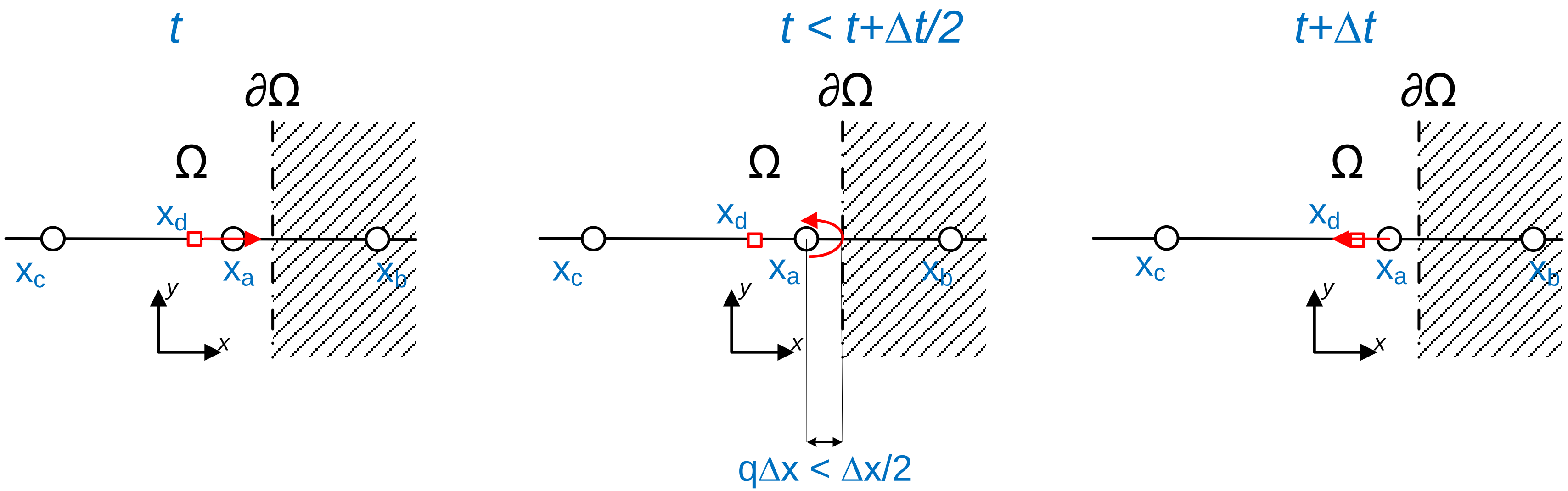
Recall the linear equation formula

$$y = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x - x_1) + y_1$$



Linear interpolation

- Use linear interpolation based on known populations:
  - → known: $\tilde{f}_R(\mathbf{x}_a, t)$
  - → known: $\tilde{f}_R(\mathbf{x}_c, t)$
  - → unknown: $\tilde{f}_R(\mathbf{x}_d, t)$

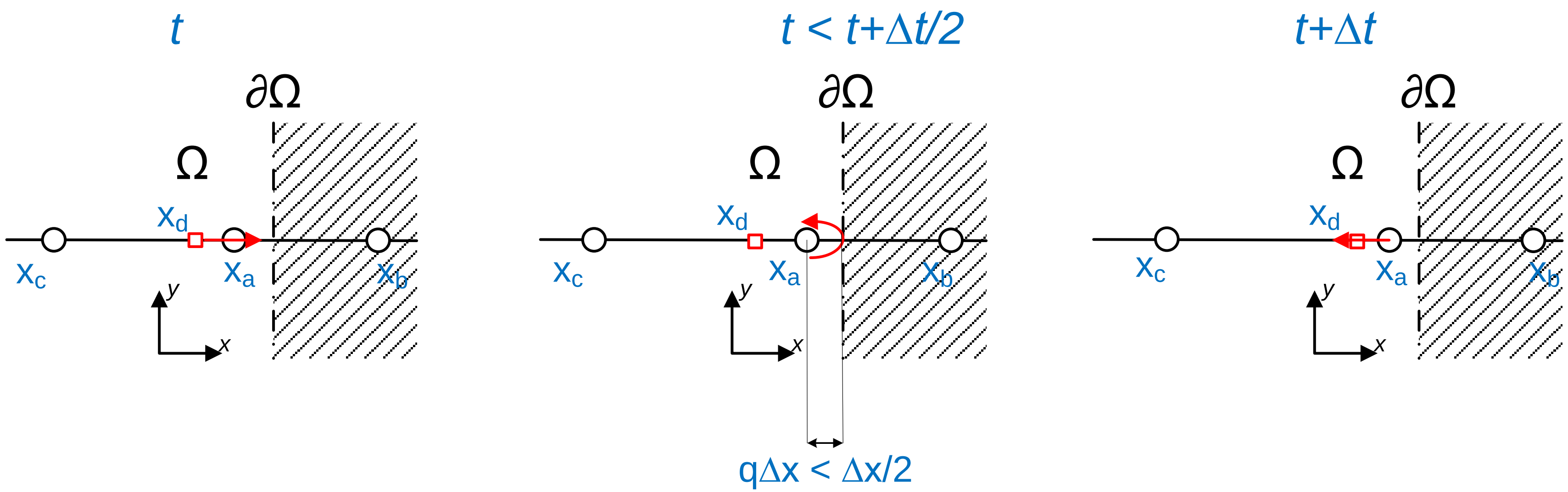


Note: $\|\mathbf{x}_c - \mathbf{x}_a\| = \|\mathbf{x}_a - \mathbf{x}_b\| = \Delta\mathbf{x}$

Linear interpolation: Exercise

Question:

Compute $\tilde{f}_R(\mathbf{x}_d, t)$ through linear interpolation of $\tilde{f}_R(\mathbf{x}_a, t)$ and $\tilde{f}_R(\mathbf{x}_c, t)$



Note: $\|\mathbf{x}_c - \mathbf{x}_a\| = \|\mathbf{x}_a - \mathbf{x}_b\| = \Delta\mathbf{x}$

Linear interpolation: Exercise

Question:

Compute $\tilde{f}_R(\mathbf{x}_d, t)$ through linear interpolation of $\tilde{f}_R(\mathbf{x}_a, t)$ and $\tilde{f}_R(\mathbf{x}_c, t)$

Solution:

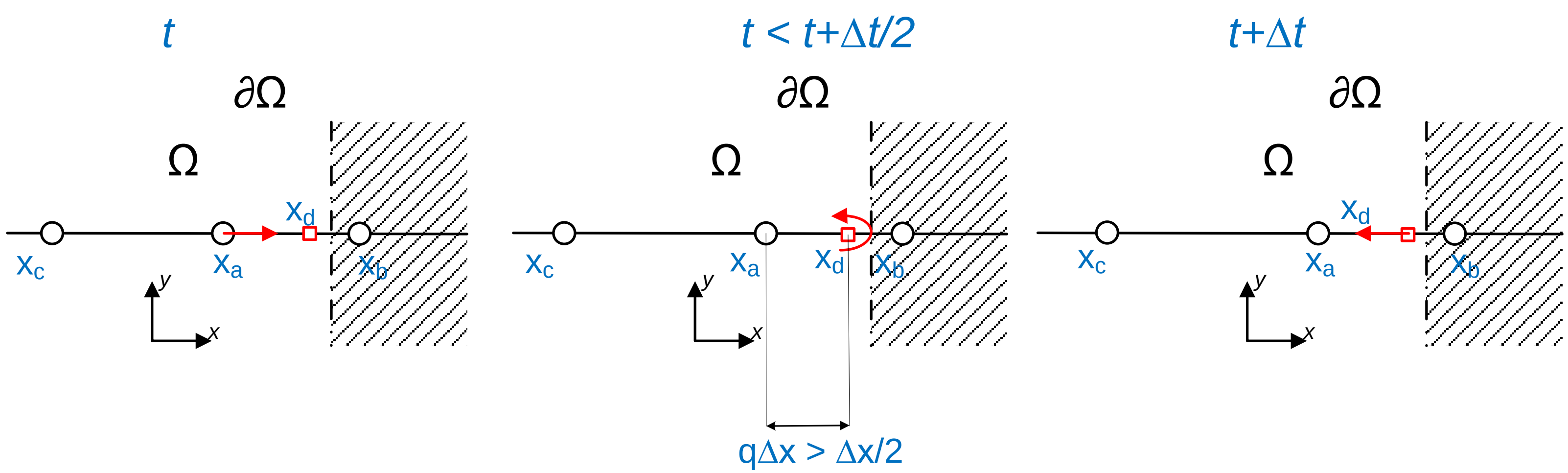
$$\tilde{f}_R(\mathbf{x}_d, t) = (1 - 2q)\tilde{f}_R(\mathbf{x}_c, t) + 2q\tilde{f}_R(\mathbf{x}_a, t)$$

that yields:

$$f_L(\mathbf{x}_a, t + \Delta t) = \tilde{f}_R(\mathbf{x}_d, t)$$

Half-way bounceback

When the **wall is beyond the half-way location**, *i.e.* $q > \frac{1}{2}$, the previous procedure leads to an extrapolation instead of an interpolation scheme

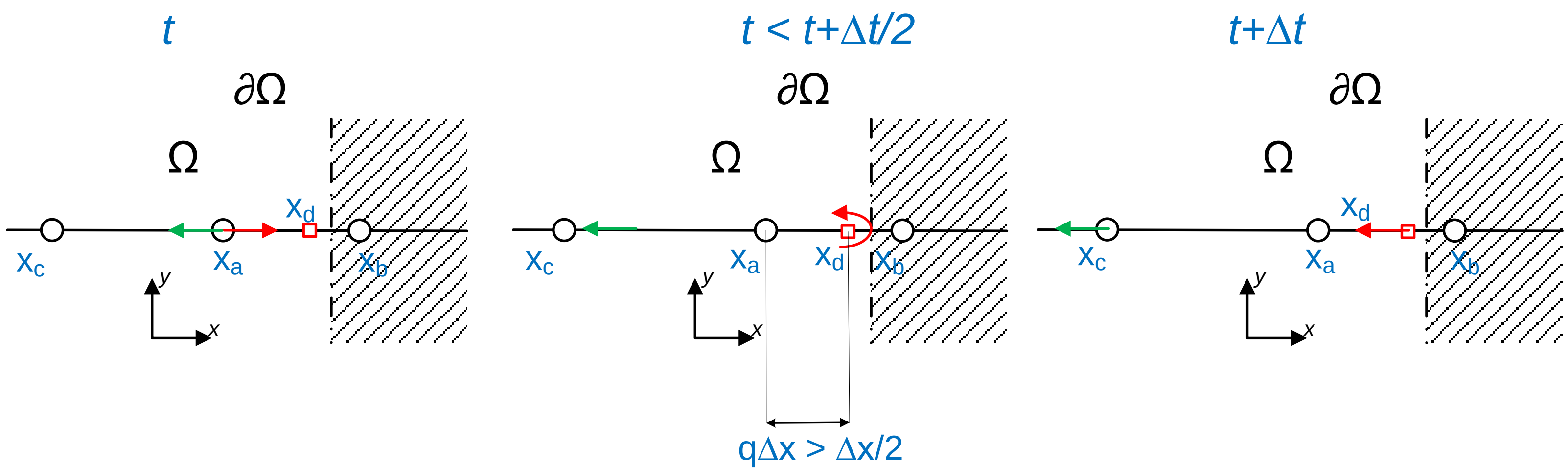


Solution:

Interpolate the post-streaming populations

Half-way bounceback

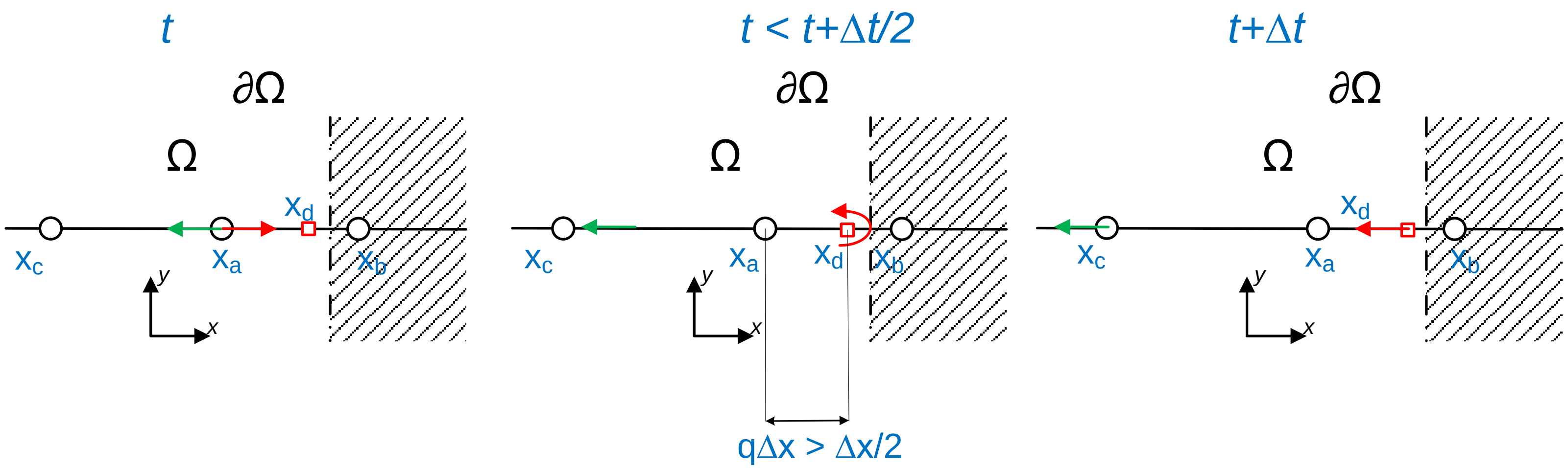
- From the time evolution sketch of the populations we observe:
  - → known: $f_L(\mathbf{x}_d, t + \Delta t) = \tilde{f}_R(\mathbf{x}_a, t)$
  - → known: $f_L(\mathbf{x}_c, t + \Delta t) = \tilde{f}_L(\mathbf{x}_a, t)$
  - → unknown: $f_L(\mathbf{x}_a, t + \Delta t)$



Linear interpolation: Exercise

Question:

Compute $f_L(\mathbf{x}_a, t+\Delta t)$ through linear interpolation of $f_L(\mathbf{x}_d, t+\Delta t)$ and $f_L(\mathbf{x}_c, t+\Delta t)$



Note: $\|\mathbf{x}_c - \mathbf{x}_a\| = \|\mathbf{x}_a - \mathbf{x}_b\| = \Delta\mathbf{x}$

Linear interpolation: Exercise

Question:

Compute $f_L(\mathbf{x}_a, t+\Delta t)$ through linear interpolation of $f_L(\mathbf{x}_d, t+\Delta t)$ and $f_L(\mathbf{x}_c, t+\Delta t)$

Solution:

$$f_L(\mathbf{x}_a, t+\Delta t) = \left(\frac{2q-1}{2q}\right) f_L(\mathbf{x}_d, t+\Delta t) + \frac{1}{2q} f_L(\mathbf{x}_d, t+\Delta t)$$

or

$$f_L(\mathbf{x}_a, t+\Delta t) = \left(\frac{2q-1}{2q}\right) \tilde{f}_R(\mathbf{x}_a, t) + \frac{1}{2q} \tilde{f}_L(\mathbf{x}_a, t)$$

Linear interpolation: 2D lattices

Linear interpolation bounceback:

- $q < \frac{1}{2}$

$$f_{\bar{\alpha}}(\mathbf{x}, t + \Delta t) = 2q\tilde{f}_{\alpha}(\mathbf{x}, t) + (1 - 2q)\tilde{f}_{\alpha}(\mathbf{x} - \mathbf{c}\Delta t, t)$$

- $q \geq \frac{1}{2}$

$$f_{\bar{\alpha}}(\mathbf{x}, t + \Delta t) = \frac{1}{2q}\tilde{f}_{\alpha}(\mathbf{x}, t) + \left(\frac{2q-1}{2q}\right)\tilde{f}_{\bar{\alpha}}(\mathbf{x}, t)$$

Quadratic interpolation: 2D lattices

Quadratic interpolation bounceback:

- $q < \frac{1}{2}$

$$f_{\bar{\alpha}}(\mathbf{x}, t + \Delta t) = q(2q+1)\tilde{f}_\alpha(\mathbf{x}, t)$$
$$+(1+2q)(1-2q)\tilde{f}_\alpha(\mathbf{x} - \mathbf{c}\Delta t, t) - q(1-2q)\tilde{f}_\alpha(\mathbf{x} - 2\mathbf{c}\Delta t, t)$$

- $q \geq \frac{1}{2}$

$$f_{\bar{\alpha}}(\mathbf{x}, t + \Delta t) = \frac{1}{q(2q+1)}\tilde{f}_\alpha(\mathbf{x}, t) + \frac{2q-1}{q}\tilde{f}_{\bar{\alpha}}(\mathbf{x}, t)$$
$$+\frac{1-2q}{1+2q}\tilde{f}_{\bar{\alpha}}(\mathbf{x} - \mathbf{c}\Delta t, t)$$

Exercise IV

Exercise IV:

Flow around circular cylinder

Outline

Curved boundary conditions in LBM

- LBM uses cartesian uniform grids
- Non-orthogonal shapes require numerical approximations, *e.g.* interpolations
- In order to preserve the advantages of LBM it is preferable to only interpolate the solution at boundary
- Bounceback provides a good framework to extend to curved boundaries through interpolations