An ADER-WENO Finite Volume AMR code for Astrophysics

Olindo Zanotti
olindo.zanotti@unitn.it

University of Trento

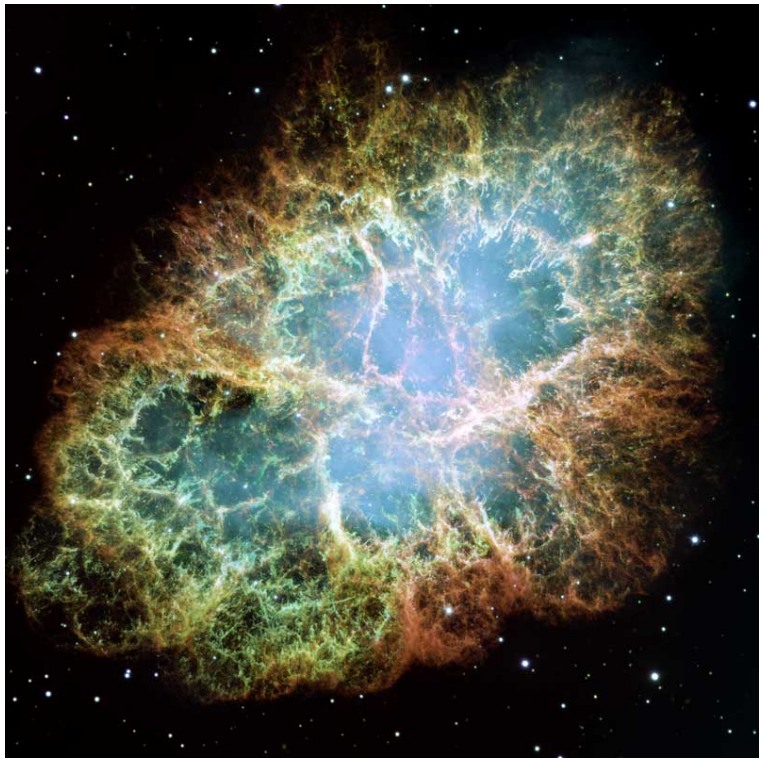
Astronum 2013
8th Intern. Conference on Numerical Modeling of Space Plasma Flows
1–5 July 2013 - Biarritz

With: M. Dumbser, A. Hidalgo, D. Balsara

Outline

Motivations

Many physical problems show great disparities in the spatial and temporal scales (**multiscale problems**), which a static grid approach cannot treat efficiently.



Adaptive Mesh Refinement (AMR) , namely the possibility to change the computational grid dynamically in space and in time, becomes necessary.

High Order Methods can also help substantially when very small details need to be solved.

- Taken separately, AMR and High Order methods have a long history

- Taken separately, AMR and High Order methods have a long history
- AMR+High Order Methods
  - Colella et al. (2009) 'High order finite-volume adaptive methods on locally rectangular grids', JPCS, 180, 012010
  - Mignone et al. (2012) 'The Pluto code for Adpative Mesh Computations in astrophysical fluid dynamics', ApJ Suppl., 198, 7
  - Burger et al. (2012) 'Spectral WENO schemes with Adaptive Mesh refinement for models of polydisperse sedimentation', ZAMM, 93, (6-7)

- Taken separately, AMR and High Order methods have a long history
- AMR+High Order Methods
  - Colella et al. (2009) 'High order finite-volume adaptive methods on locally rectangular grids', JPCS, 180, 012010
  - Mignone et al. (2012) 'The Pluto code for Adpative Mesh Computations in astrophysical fluid dynamics', ApJ Suppl., 198, 7
  - Burger et al. (2012) 'Spectral WENO schemes with Adaptive Mesh refinement for models of polydisperse sedimentation', ZAMM, 93, (6-7)
- A particularly appealing implementation of High order methods is represented by ADER schemes, which are one-step time-update schemes.
  - Original version: use the Lax-Wendroff procedure [Toro et al. (2001); Titarev and Toro (2002) ...]
  - Modern version: use a weak integral formulation of the governing PDE [Dumbser et al. (2008) JCP, 227, 3971]

- Taken separately, AMR and High Order methods have a long history
- AMR+High Order Methods
  - Colella et al. (2009) 'High order finite-volume adaptive methods on locally rectangular grids', JPCS, 180, 012010
  - Mignone et al. (2012) 'The Pluto code for Adpative Mesh Computations in astrophysical fluid dynamics', ApJ Suppl., 198, 7
  - Burger et al. (2012) 'Spectral WENO schemes with Adaptive Mesh refinement for models of polydisperse sedimentation', ZAMM, 93, (6-7)
- A particularly appealing implementation of High order methods is represented by ADER schemes, which are one-step time-update schemes.
  - Original version: use the Lax-Wendroff procedure [Toro et al. (2001); Titarev and Toro (2002) ...]
  - Modern version: use a weak integral formulation of the governing PDE [Dumbser et al. (2008) JCP, 227, 3971]
- AMR+ADER schemes.... presented very recently by
  - Dumbser, Zanotti, Hidalgo, Balsara (2013) 'ADER-WENO Finite Volume schemes with space-time adaptive mesh refinement' JCP, 248, 257-286

Outline

Finite volume scheme

We consider **hyperbolic systems** of balance laws in Cartesian coordinates

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} + \frac{\partial \mathbf{h}}{\partial z} = \mathbf{S}(\mathbf{u}, \mathbf{x}, t)$$

We use a **Standard Finite Volume discretization**

$$\mathbf{u}_{ijk}^{n+1} = \mathbf{u}_{ijk}^{n} - \frac{\Delta t}{\Delta x_i}\left(\mathbf{f}_{i+\frac{1}{2},jk} - \mathbf{f}_{i-\frac{1}{2},jk}\right) - \frac{\Delta t}{\Delta y_j}\left(\mathbf{g}_{i,j+\frac{1}{2},k} - \mathbf{g}_{i,j-\frac{1}{2},k}\right) - \frac{\Delta t}{\Delta z_k}\left(\mathbf{h}_{ij,k+\frac{1}{2}} - \mathbf{h}_{ij,k-\frac{1}{2}}\right) + \Delta t \mathbf{S}_{ijk},$$

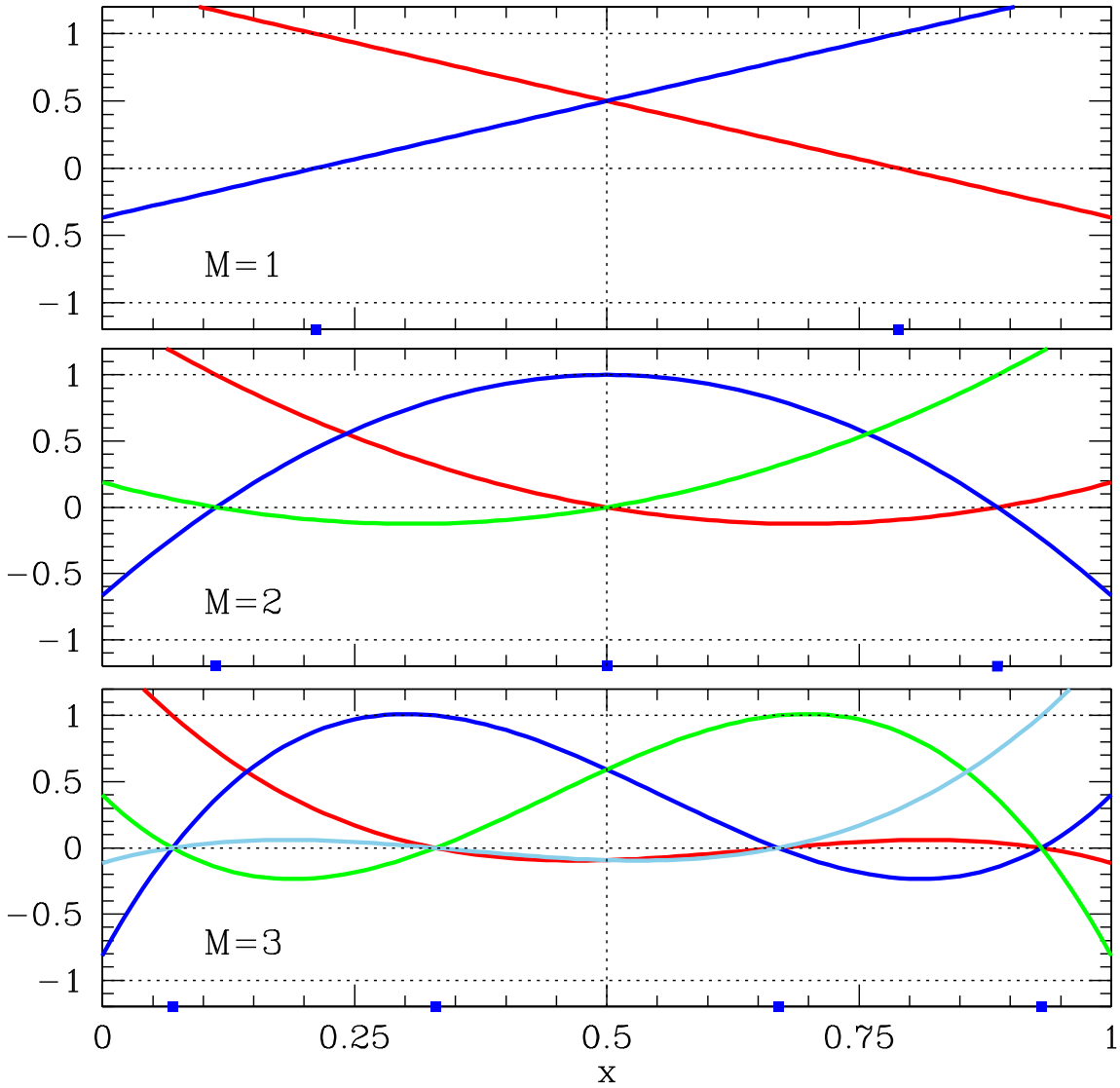
over the control volumes $I_{ijk} = [x_{i-\frac{1}{2}}; x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}; y_{j+\frac{1}{2}}] \times [z_{k-\frac{1}{2}}; z_{k+\frac{1}{2}}]$, with

$$\mathbf{u}_{ijk}^{n} = \frac{1}{\Delta x_i}\frac{1}{\Delta y_j}\frac{1}{\Delta z_k}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}\int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}} \mathbf{u}(x, y, z, t^n)\, dz\, dy\, dx$$

$$\mathbf{f}_{i+\frac{1}{2},jk} = \frac{1}{\Delta t}\frac{1}{\Delta y_j}\frac{1}{\Delta z_k}\int_{t^n}^{t^{n+1}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}\int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}} \tilde{\mathbf{f}}\left(\mathbf{q}_h^{-}(x_{i+\frac{1}{2}}, y, z, t), \mathbf{q}_h^{+}(x_{i+\frac{1}{2}}, y, z, t)\right) dz\, dy\, dt,$$

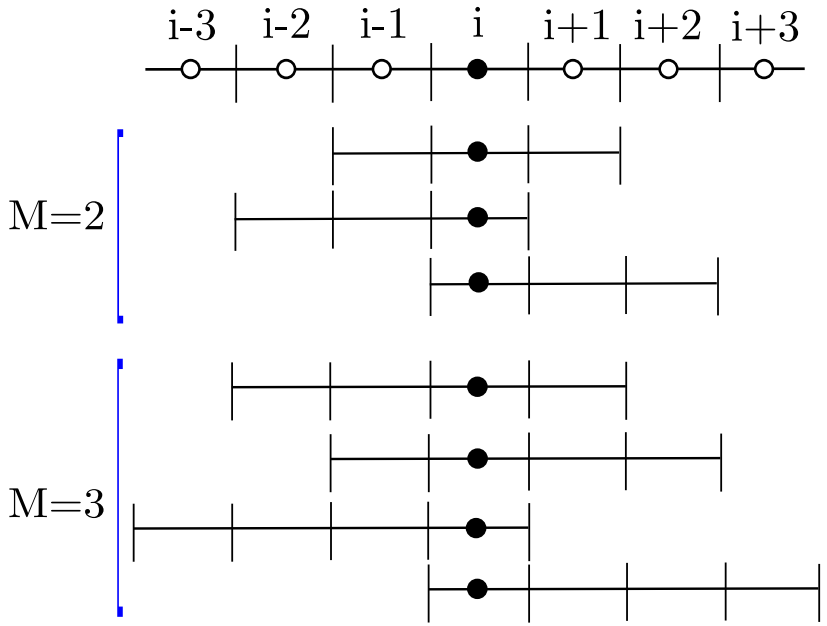
$$\mathbf{S}_{ijk} = \frac{1}{\Delta t}\frac{1}{\Delta x_i}\frac{1}{\Delta y_j}\frac{1}{\Delta z_k}\int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}\int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}} \mathbf{S}\left(\mathbf{q}_h(x, y, z, t)\right) dz\, dy\, dx\, dt\,.$$

- Introduce a **nodal basis** of polynomials $\psi_l(\xi)$ of order M defined with respect to a set of Gauss-Legendre nodal points λ_k, such that $\psi_l(\lambda_k) = \delta_{lk}$



- Build **one–dimensional reconstruction stencils** along each direction

$$\mathcal{S}_{ijk}^{s,x} = \bigcup_{e=i-L}^{i+R} I_{ejk}, \quad \mathcal{S}_{ijk}^{s,y} = \bigcup_{e=j-L}^{j+R} I_{iek}, \quad \mathcal{S}_{ijk}^{s,z} = \bigcup_{e=k-L}^{k+R} I_{ije},$$



- three stencils for even M
- four stencils for odd M
- $M+1$ cells in each stencil

1. Perform an entire polynomial **WENO reconstuction** along $x-$ **direction**:

$$\mathbf{w}_h^{s,x}(x,t^n) = \sum_{p=0}^{M} \psi_p(\xi)\hat{\mathbf{w}}_{ijk,p}^{n,s} := \psi_p(\xi)\hat{\mathbf{w}}_{ijk,p}^{n,s} \qquad \forall \qquad \mathcal{S}_{ijk}^{s,x}$$

Impose integral conservation on all elements of the stencil:

$$\frac{1}{\Delta x_e}\int_{x_{e-\frac{1}{2}}}^{x_{e+\frac{1}{2}}} \psi_p(\xi(x))\hat{\mathbf{w}}_{ijk,p}^{n,s}\, dx = \mathbf{u}_{ejk}^n, \qquad \forall l_{ejk} \in \mathcal{S}_{ijk}^{s,x}$$

Perform a data-dependent nonlinear combination:

$$\mathbf{w}_h^x(x,t^n) = \psi_p(\xi)\hat{\mathbf{w}}_{ijk,p}^n, \qquad \text{with} \qquad \hat{\mathbf{w}}_{ijk,p}^n = \sum_{s=1}^{N_s} \omega_s \hat{\mathbf{w}}_{ijk,p}^{n,s}$$

$$\omega_s = \frac{\tilde{\omega}_s}{\sum_k \tilde{\omega}_k}, \qquad \tilde{\omega}_s = \frac{\lambda_s}{(\sigma_s+\epsilon)^r}$$

$$\sigma_s = \Sigma_{pm}\hat{\mathbf{w}}_{ijk,p}^{n,s}\hat{\mathbf{w}}_{ijk,m}^{n,s} \qquad \Sigma_{pm} = \sum_{\alpha=1}^{M}\int_0^1 \frac{\partial^\alpha \psi_p(\xi)}{\partial \xi^\alpha}\cdot\frac{\partial^\alpha \psi_m(\xi)}{\partial \xi^\alpha} d\xi\,.$$

[see Dumbser & Käser, (2007), JCP, 221, 693]

2. Perform a second polynomial **WENO reconstuction** along $y-$ **direction** using as input the $M+1$ degrees of freedom $\hat{\mathbf{w}}_{ijk,p}^{n}$:

$$\mathbf{w}_h^{s,y}(x,y,t^n) = \psi_p(\xi)\psi_q(\eta)\hat{\mathbf{w}}_{ijk,pq}^{n,s}\,.$$

Apply integral conservation is now in the y direction:

$$\frac{1}{\Delta y_e}\int_{y_{e-\frac{1}{2}}}^{y_{e+\frac{1}{2}}} \psi_q(\eta(y))\hat{\mathbf{w}}_{ijk,pq}^{n,s}\,dy = \hat{\mathbf{w}}_{iek,p}^{n}, \qquad \forall I_{iek}\in\mathcal{S}_{ijk}^{s,y}\,.$$

$\Longrightarrow$

$$\mathbf{w}_h^{y}(x,y,t^n) = \psi_p(\xi)\psi_q(\eta)\hat{\mathbf{w}}_{ijk,pq}^{n}, \qquad \text{with} \qquad \hat{\mathbf{w}}_{ijk,pq}^{n} = \sum_{s=1}^{N_s}\omega_s\hat{\mathbf{w}}_{ijk,pq}^{n,s},$$

3. Perform the last polynomial **WENO reconstuction** along $z-$ **direction** using as input the $(M+1)^2$ degrees of freedom $\hat{\mathbf{w}}^{n}_{ijk,pq}$: We therefore have

$$\mathbf{w}_h^{s,z}(x,y,z,t^n) = \psi_p(\xi)\psi_q(\eta)\psi_r(\zeta)\hat{\mathbf{w}}^{n,s}_{ijk,pqr}\,.$$

with the integral conservation written as above,

$$\frac{1}{\Delta z_e}\int_{z_{e-\frac{1}{2}}}^{z_{e+\frac{1}{2}}} \psi_r(\zeta(z))\hat{\mathbf{w}}^{n,s}_{ijk,pqr}\,dz = \hat{\mathbf{w}}^{n}_{iek,pq}, \qquad \forall l_{ije} \in \mathcal{S}^{s,z}_{ijk}.$$

$\Longrightarrow$ The final three-dimensional WENO polynomial

$$\mathbf{w}_h(\mathbf{x},t^n) = \psi_p(\xi)\psi_q(\eta)\psi_r(\zeta)\hat{\mathbf{w}}^{n}_{ijk,pqr}, \qquad \hat{\mathbf{w}}^{n}_{ijk,pqr} = \sum_{s=1}^{N_s}\omega_s\hat{\mathbf{w}}^{n,s}_{ijk,pqr}.$$

Outline

Local space-time DG predictor [see Dumbser et al. (2008), JCP, 227, 3971]

An alternative to the Cauchy-Kovalewski procedure to obtain the time evolution of the reconstructed polynomials and build one-step time-update numerical schemes.
In practice we need

$$\mathbf{q}_h \Longrightarrow \mathbf{f}_{i+\frac{1}{2},jk} = \frac{1}{\Delta t}\frac{1}{\Delta y_j}\frac{1}{\Delta z_k}\int_{t^n}^{t^{n+1}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}\int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}} \tilde{\mathbf{f}}\left(\mathbf{q}_h^-(x_{i+\frac{1}{2}},y,z,t),\mathbf{q}_h^+(x_{i+\frac{1}{2}},y,z,t)\right) dz\, dy, dt,$$

$$\frac{\partial \mathbf{u}}{\partial \tau} + \frac{\partial \mathbf{f}^*}{\partial \xi} + \frac{\partial \mathbf{g}^*}{\partial \eta} + \frac{\partial \mathbf{h}^*}{\partial \zeta} = \mathbf{S}^*$$

with

$$\mathbf{f}^* = \frac{\Delta t}{\Delta x_i}\mathbf{f}, \quad \mathbf{g}^* = \frac{\Delta t}{\Delta y_j}\mathbf{g}, \quad \mathbf{h}^* = \frac{\Delta t}{\Delta z_k}\mathbf{h}, \quad \mathbf{S}^* = \Delta t \mathbf{S}.$$

We then multiply by the test function $\theta_p(\boldsymbol{\xi},\tau)$ and integrate in space-time

$$\int_0^1\int_0^1\int_0^1\int_0^1 \theta_q \left(\frac{\partial \mathbf{u}}{\partial \tau} + \frac{\partial \mathbf{f}^*}{\partial \xi} + \frac{\partial \mathbf{g}^*}{\partial \eta} + \frac{\partial \mathbf{h}^*}{\partial \zeta} - \mathbf{S}^*\right) d\xi d\eta d\zeta d\tau = 0.$$

where we choose a tensor product of the basis functions
$\theta_p(\boldsymbol{\xi},\tau) = \psi_p(\xi)\psi_q(\eta)\psi_r(\zeta)\psi_s(\tau)$.

Integration by parts in time yields...

$$\int_0^1\int_0^1\int_0^1 \theta_q(\boldsymbol{\xi},1)\mathbf{u}(\boldsymbol{\xi},1)d\xi d\eta d\zeta - \int_0^1\int_0^1\int_0^1\int_0^1 \left(\frac{\partial}{\partial\tau}\theta_q\right)\mathbf{u}d\xi d\eta d\zeta d\tau$$
$$+\int_0^1\int_0^1\int_0^1\int_0^1 \left[\theta_q\left(\frac{\partial \mathbf{f}^*}{\partial\xi}+\frac{\partial \mathbf{g}^*}{\partial\eta}+\frac{\partial \mathbf{h}^*}{\partial\zeta}-\mathbf{S}^*\right)\right]d\xi d\eta d\zeta d\tau$$
$$=\int_0^1\int_0^1\int_0^1 \theta_q(\boldsymbol{\xi},0)\mathbf{w}_h(\boldsymbol{\xi},t^n)d\xi d\eta d\zeta.$$

We introduce the discrete space-time solution $\mathbf{q}_h$

$$\mathbf{q}_h = \mathbf{q}_h(\boldsymbol{\xi},\tau) = \theta_p\left(\boldsymbol{\xi},\tau\right)\hat{\mathbf{q}}_p,$$

and similarly for the fluxes and sources

$$\mathbf{f}_h^* = \theta_p\hat{\mathbf{f}}_p^* = \theta_p\mathbf{f}^*\left(\hat{\mathbf{q}}_p\right), \qquad \ldots \qquad \mathbf{S}_h^* = \theta_p\hat{\mathbf{S}}_p^* = \theta_p\mathbf{S}^*\left(\hat{\mathbf{q}}_p\right).$$

Insert everything in Eq. above and obtain....

$$\int_0^1\int_0^1\int_0^1 \theta_q(\boldsymbol{\xi},1)\theta_p(\boldsymbol{\xi},1)\hat{\mathbf{q}}_p \, d\xi d\eta d\zeta - \int_0^1\int_0^1\int_0^1\int_0^1 \left(\frac{\partial}{\partial\tau}\theta_q\right)\theta_p\hat{\mathbf{q}}_p \, d\xi d\eta d\zeta d\tau$$

$$+\int_0^1\int_0^1\int_0^1\int_0^1 \left[\theta_q\left(\frac{\partial}{\partial\xi}\theta_p\hat{\mathbf{f}}_p^* + \frac{\partial}{\partial\eta}\theta_p\hat{\mathbf{g}}_p^* + \frac{\partial}{\partial\zeta}\theta_p\hat{\mathbf{h}}_p^* - \theta_p\hat{\mathbf{S}}_p^*\right)\right] d\xi d\eta d\zeta d\tau$$

$$= \int_0^1\int_0^1\int_0^1 \theta_q(\boldsymbol{\xi},0)\mathbf{w}_h(\boldsymbol{\xi},t^n)\, d\xi d\eta d\zeta\,,$$

which is a nonlinear algebraic equation system for the unknown coefficients $\hat{\mathbf{q}}_p$.

In a more compact form:

$$\mathbf{K}^1_{qp}\hat{\mathbf{q}}_p + \mathbf{K}^\xi_{qp}\cdot\hat{\mathbf{f}}_p^* + \mathbf{K}^\eta_{qp}\hat{\mathbf{g}}_p^* + \mathbf{K}^\zeta_{qp}\hat{\mathbf{h}}_p^* = \mathbf{M}_{qp}\hat{\mathbf{S}}_p^* + \mathbf{F}^0_{qm}\hat{\mathbf{w}}^n_m,$$

with the various matrices defined as

$$\mathbf{K}^1_{qp} = \int_0^1\int_0^1\int_0^1 \theta_q(\boldsymbol{\xi},1)\theta_p(\boldsymbol{\xi},1)d\boldsymbol{\xi} - \int_0^1\int_0^1\int_0^1\int_0^1 \left(\frac{\partial}{\partial\tau}\theta_q\right)\theta_p\, d\boldsymbol{\xi}d\tau,$$

$$\mathbf{K}^{\boldsymbol{\xi}}_{qp} = \left(\mathbf{K}^\xi_{qp},\mathbf{K}^\eta_{qp},\mathbf{K}^\zeta_{qp}\right) = \int_0^1\int_0^1\int_0^1\int_0^1 \theta_q\frac{\partial}{\partial\boldsymbol{\xi}}\theta_p\, d\boldsymbol{\xi}d\tau,$$

$$\mathbf{M}_{qp} = \int_0^1\int_0^1\int_0^1\int_0^1 \theta_q\theta_p\, d\boldsymbol{\xi}d\tau, \qquad \mathbf{F}^0_{qp} = \int_0^1\int_0^1\int_0^1 \theta_q(\boldsymbol{\xi},0)\psi_m(\boldsymbol{\xi})d\boldsymbol{\xi},$$

The product of the matrices with the vectors of degrees of freedom can be efficiently implemented in a dimension–by–dimension manner. An iterative scheme can be adopted

[see Dumbser & Zanotti (2009), JCP, 228, 6991]

$$\mathbf{K}^{1}_{qp}\hat{\mathbf{q}}^{k+1}_{p} - \mathbf{M}_{qp}\hat{\mathbf{S}}^{*,k+1}_{p} = \mathbf{F}^{0}_{qm}\hat{\mathbf{w}}^{n}_{m} - \mathbf{K}^{\xi}_{qp}\cdot\hat{\mathbf{f}}^{*,k}_{p} - \mathbf{K}^{\eta}_{qp}\hat{\mathbf{g}}^{*,k}_{p} - \mathbf{K}^{\zeta}_{qp}\hat{\mathbf{h}}^{*,k}_{p}$$

Once this is done, we have everything to write the scheme as

$$\begin{aligned}\mathbf{u}^{n+1}_{ijk} &= \mathbf{u}^{n}_{ijk} - \frac{\Delta t}{\Delta x_i}\left(\mathbf{f}_{i+\frac{1}{2},jk} - \mathbf{f}_{i-\frac{1}{2},jk}\right) - \frac{\Delta t}{\Delta y_j}\left(\mathbf{g}_{i,j+\frac{1}{2},k} - \mathbf{g}_{i,j-\frac{1}{2},k}\right)\\ &\quad - \frac{\Delta t}{\Delta z_k}\left(\mathbf{h}_{ij,k+\frac{1}{2}} - \mathbf{h}_{ij,k-\frac{1}{2}}\right) + \Delta t\mathbf{S}_{ijk},\end{aligned}$$

- Local Lax-Friedrichs flux

$$\tilde{\mathbf{f}}\left(\mathbf{q}^{-}_{h},\mathbf{q}^{+}_{h}\right) = \frac{1}{2}\left(\mathbf{f}(\mathbf{q}^{-}_{h}) + \mathbf{f}(\mathbf{q}^{+}_{h})\right) - \frac{1}{2}|s_{\max}|\left(\mathbf{q}^{+}_{h} - \mathbf{q}^{-}_{h}\right), \qquad (1)$$

  where $|s_{\max}|$ denotes the maximum absolute value of the eigenvalues of the Jacobian matrix $\mathbf{A} = \partial\mathbf{f}/\partial\mathbf{u}$.

- Osher–type flux

$$\tilde{\mathbf{f}}\left(\mathbf{q}^{-}_{h},\mathbf{q}^{+}_{h}\right) = \frac{1}{2}\left(\mathbf{f}(\mathbf{q}^{-}_{h}) + \mathbf{f}(\mathbf{q}^{+}_{h})\right) - \frac{1}{2}\left(\int_0^1 |\mathbf{A}(\psi(s))|\,ds\right)\left(\mathbf{q}^{+}_{h} - \mathbf{q}^{-}_{h}\right), \qquad (2)$$

Outline

AMR implementation

We have developed a **cell-by-cell** AMR technique in which the computational domain is discretized with a uniform Cartesian grid at the coarsest level.

- Adopt a refinement criterion, marking a cell for refinement if $\chi_m > \chi_{\rm ref}$, where

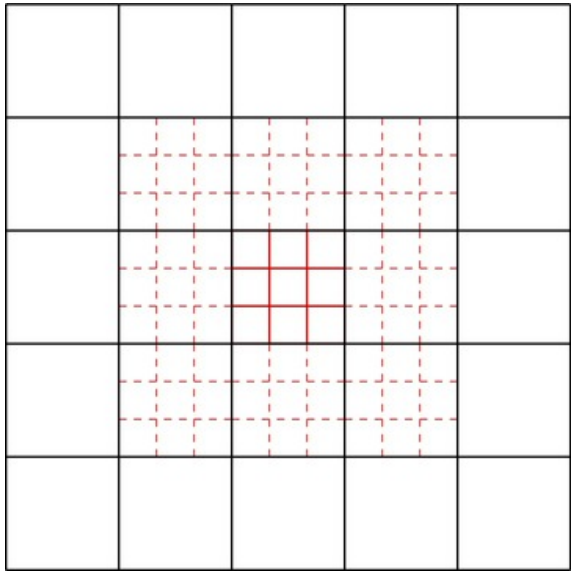
$$\chi_m = \sqrt{\frac{\sum_{k,l}(\partial^2\Phi/\partial x_k \partial x_l)^2}{\sum_{k,l}[(|\partial\Phi/\partial x_k|_{i+1} + |\partial\Phi/\partial x_k|_i)/\Delta x_l + \varepsilon|(\partial^2/\partial x_k \partial x_l)||\Phi|]^2}}.$$

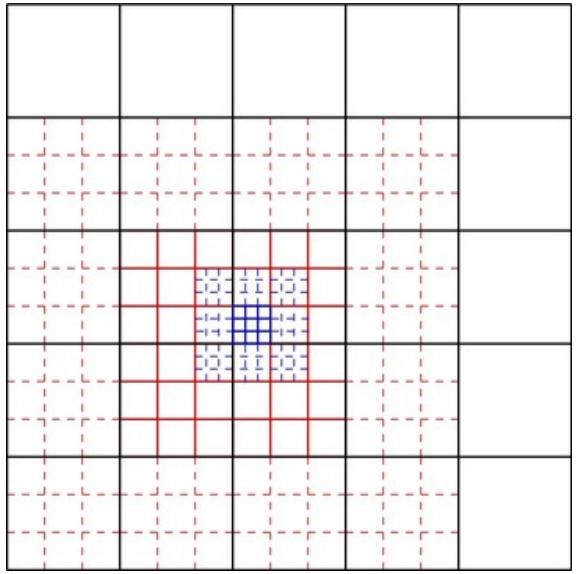
- When a cell of the level ℓ is refined, it is subdivided as

$$\Delta x_\ell = \mathfrak{r}\Delta x_{\ell+1} \quad \Delta y_\ell = \mathfrak{r}\Delta y_{\ell+1} \quad \Delta z_\ell = \mathfrak{r}\Delta z_{\ell+1} \quad \Delta t_\ell = \mathfrak{r}\Delta t_{\ell+1}$$

- Each cell $\mathcal{C}_m$, at any level of refinement, has one among three possible *status* flags.
  - *active cell*
  - *virtual child cell*
  - *virtual mother cell*

...AMR implementation





- If the Voronoi neighbors of an active refined cell $\mathcal{C}_m$ are not themselves at the same level of refinement of $\mathcal{C}_m$, they have virtual children at the same level of refinement of $\mathcal{C}_m$.
- In order to keep the reconstruction local on the coarser grid level, we have $\mathfrak{r} \geq M$.
- The levels of refinement of two cells that are Voronoi neighbors of each other can only differ by at most unity.

- **The beauty of the local-spacetime DG predictor**: it does not need any exchange of information with neighbor elements, even if two adjacent cells are on different levels of refinement.
- **Projection**

  Projection is the typical AMR operation, by which an active mother assigns values to the virtual children ($\sigma = 1$) at intermediate times

$$\bar{\mathbf{u}}_m(t_\ell^n) = \frac{1}{\Delta x_\ell}\frac{1}{\Delta y_\ell}\frac{1}{\Delta z_\ell}\int_{\mathcal{C}_m} \mathbf{q}_h(\mathbf{x}, t_\ell^n) d\mathbf{x}. \tag{3}$$

  Needed for performing the reconstruction on the finer grid level at intermediate times.
- **Averaging**

  Averaging is another typical AMR operation by which a virtual mother cell ($\sigma = -1$) obtains its cell average by averaging recursively over the cell averages of all its children at higher refinement levels.

$$\bar{\mathbf{u}}_m = \frac{1}{\mathfrak{r}^d}\sum_{\mathcal{C}_k \in \mathcal{B}_m} \bar{\mathbf{u}}_k. \tag{4}$$

Outline

Local Timestepping

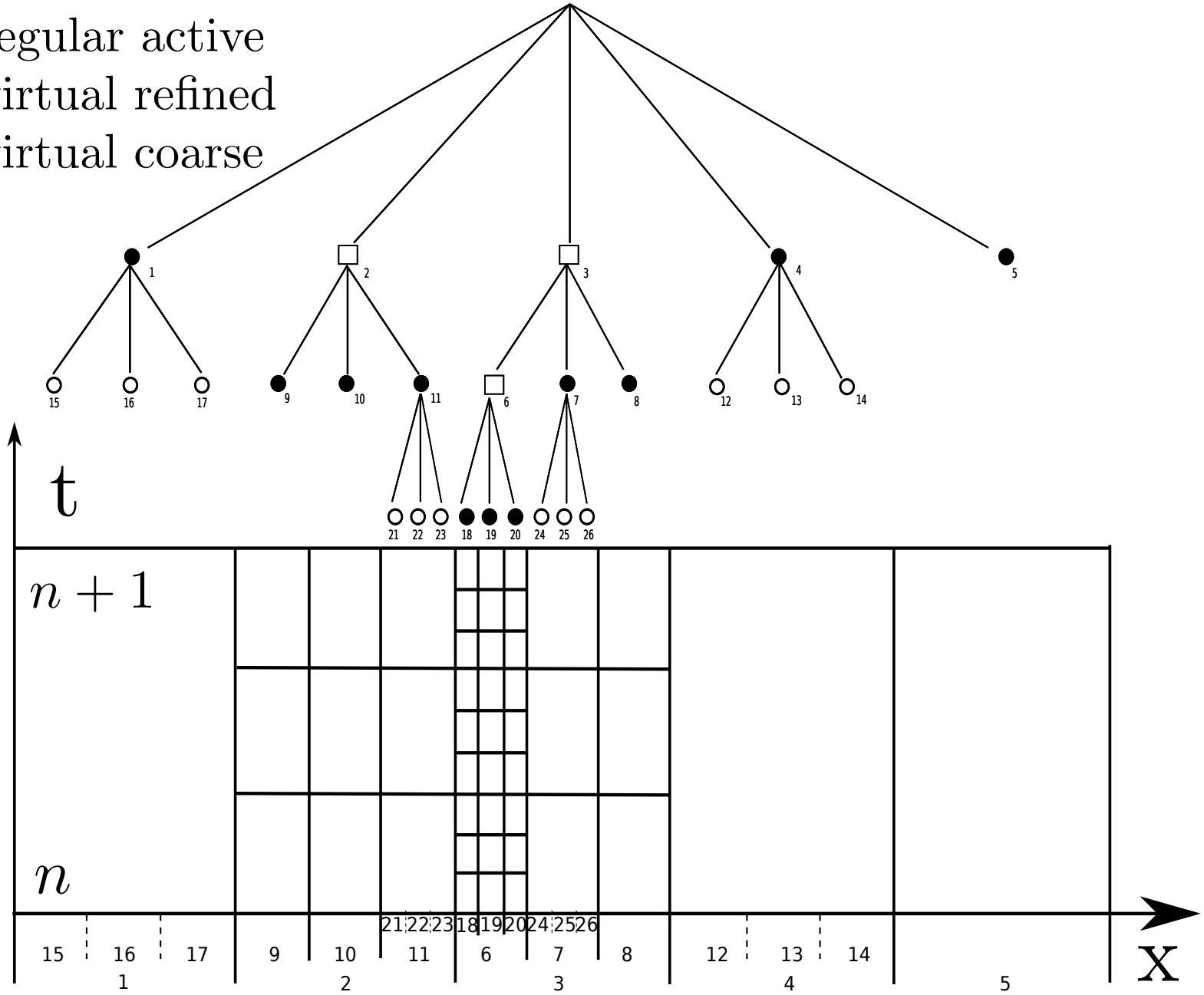
Every refinement level is advanced in time with its local timestep $\Delta t_\ell = \mathfrak{r}\Delta t_{\ell+1}$. **Update criterion:**

$$t_\ell^{n+1} \leq t_{\ell-1}^{n+1}, \qquad 0 \leq \ell \leq \ell_{\max}, \tag{5}$$

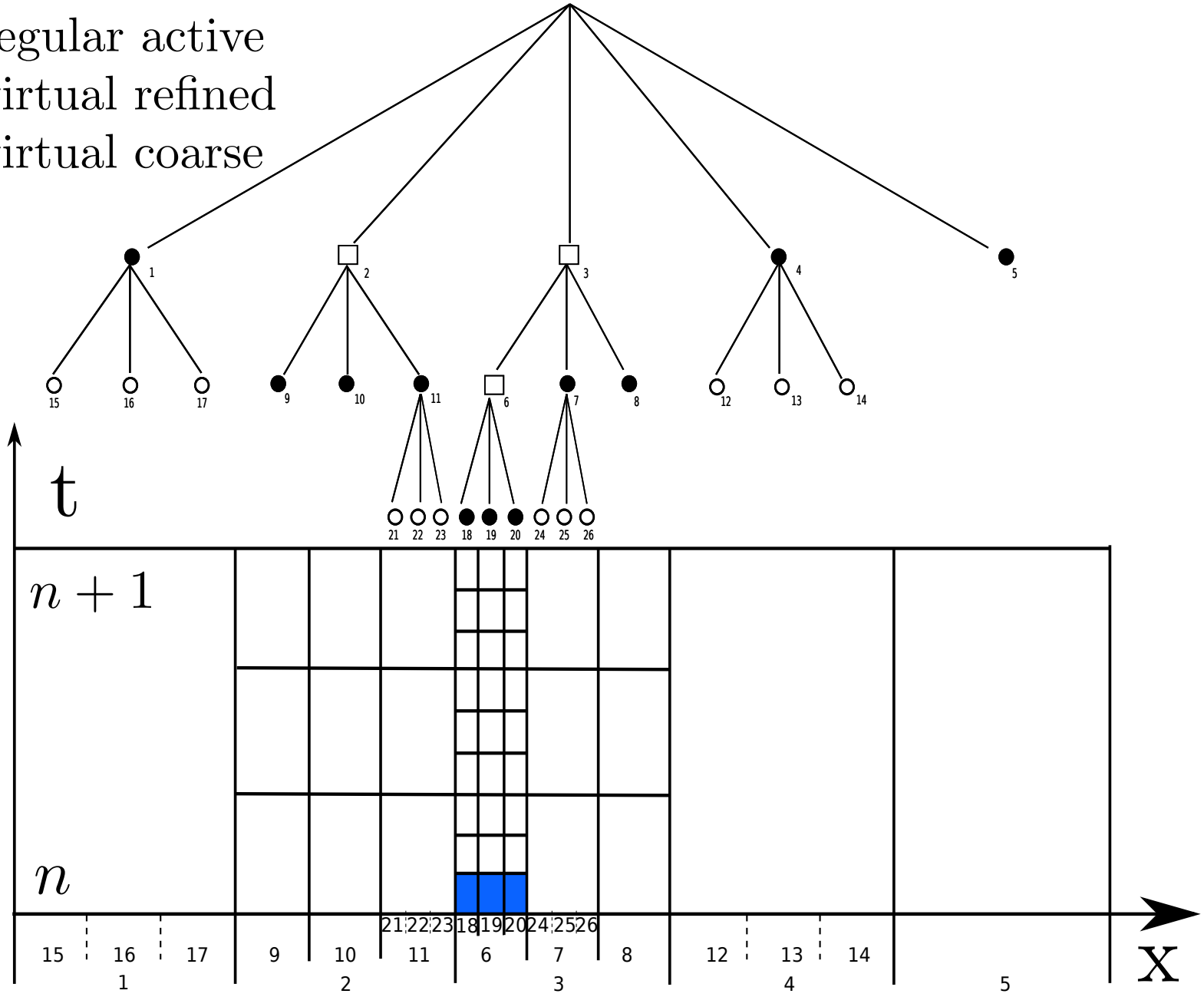
Starting from the common initial time $t = 0$, the finest level of refinement $\ell_{\max}$ is evolved first and performs a number of $\mathfrak{r}$ sub-timesteps before the next coarser level $\ell_{\max} - 1$ performs its first time update.

$\Longrightarrow$ A total amount of $\mathfrak{r}^\ell$ sub-timesteps on each level are performed in order to reach the time t_0^{n+1} of the coarsest level.

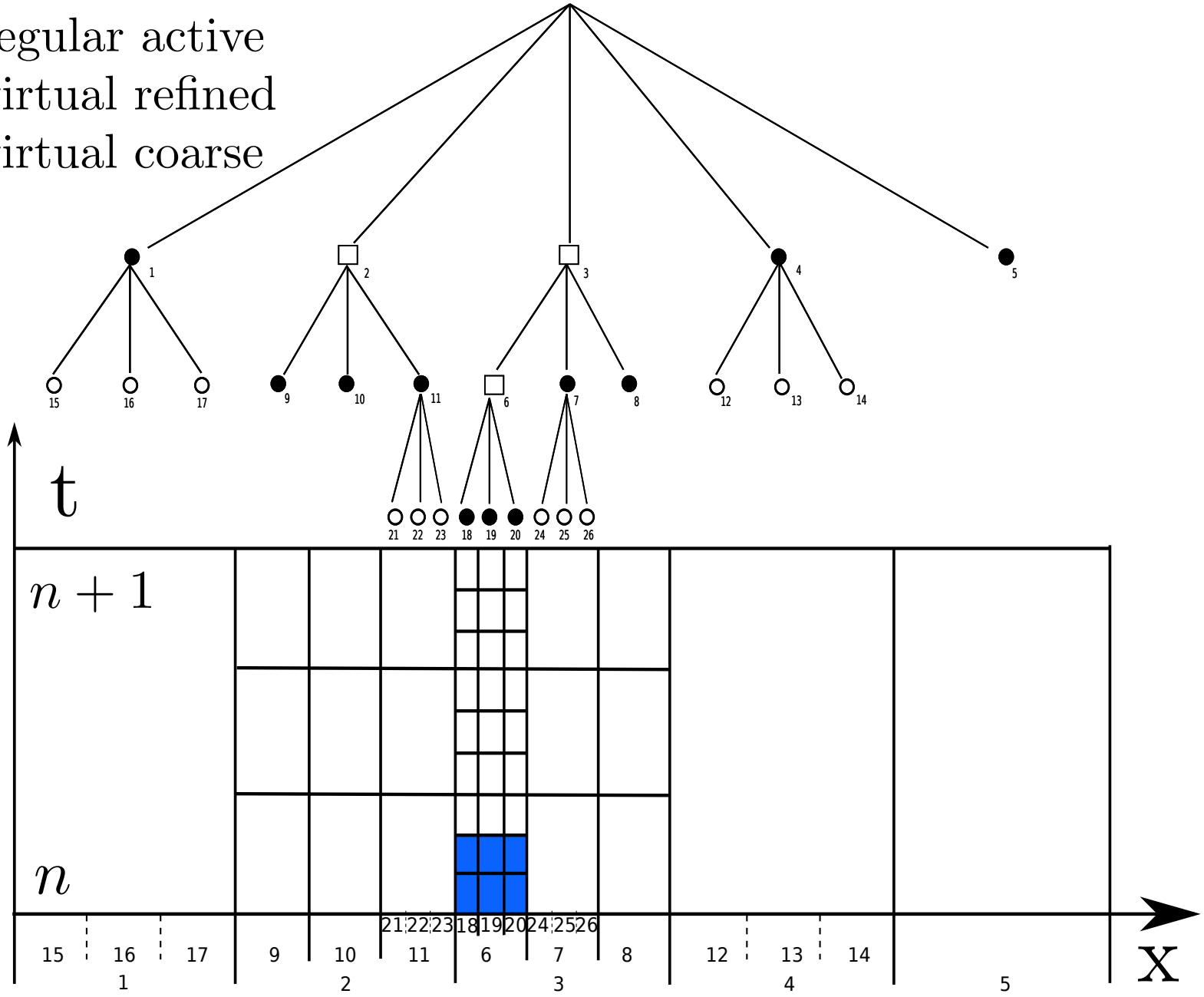
- regular active
- virtual refined
- virtual coarse



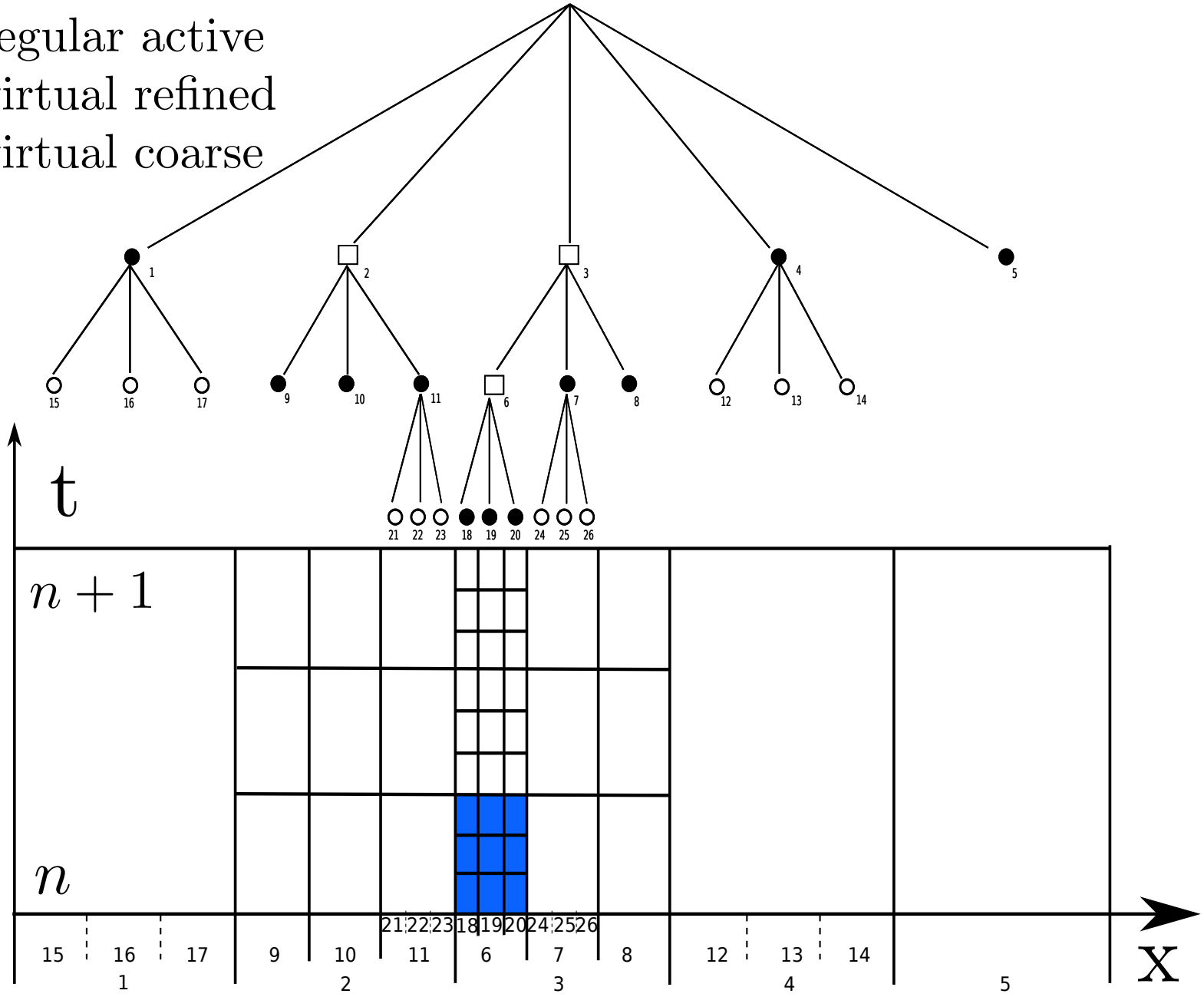
- regular active
- virtual refined
- virtual coarse



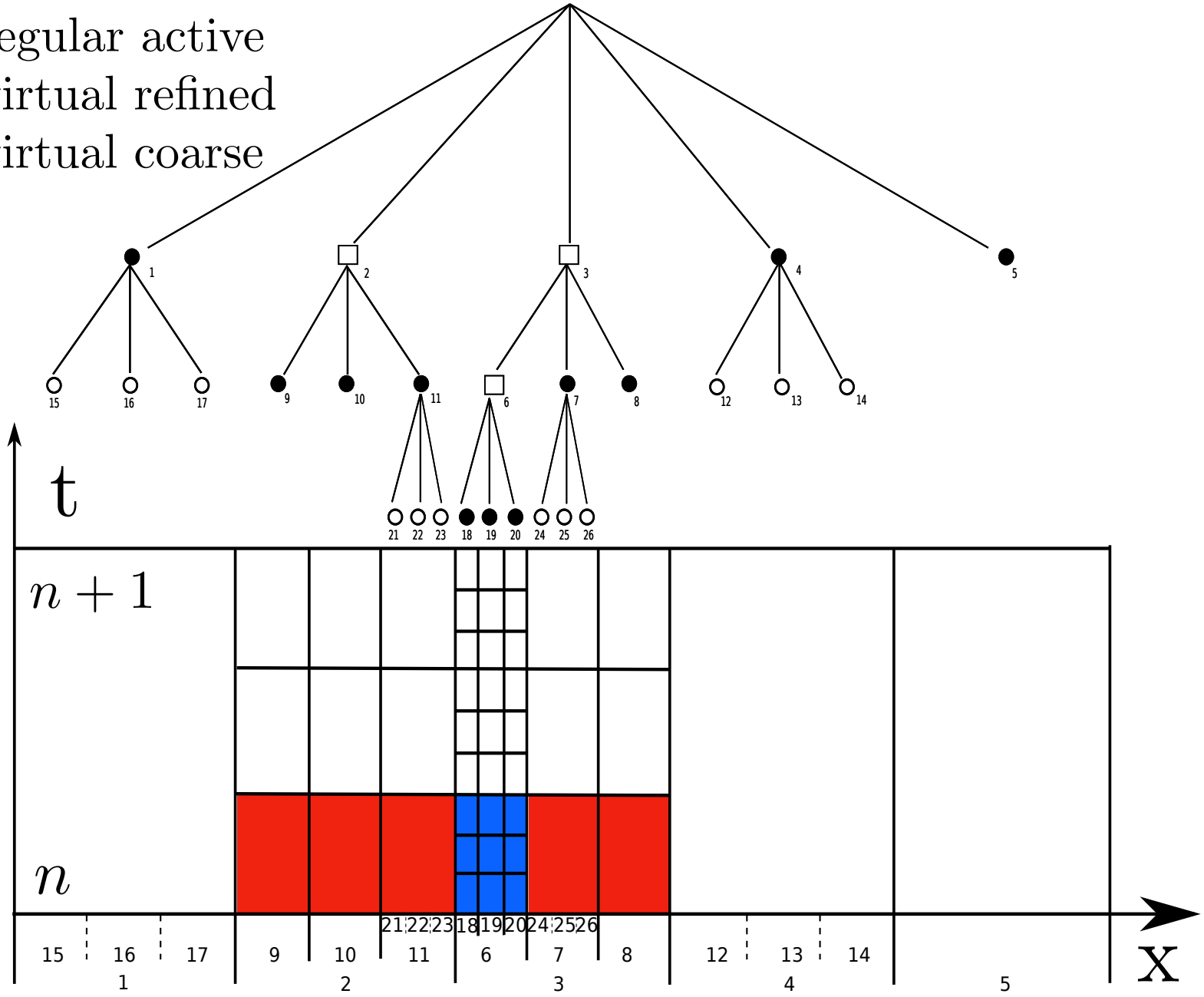
- regular active
- virtual refined
- virtual coarse



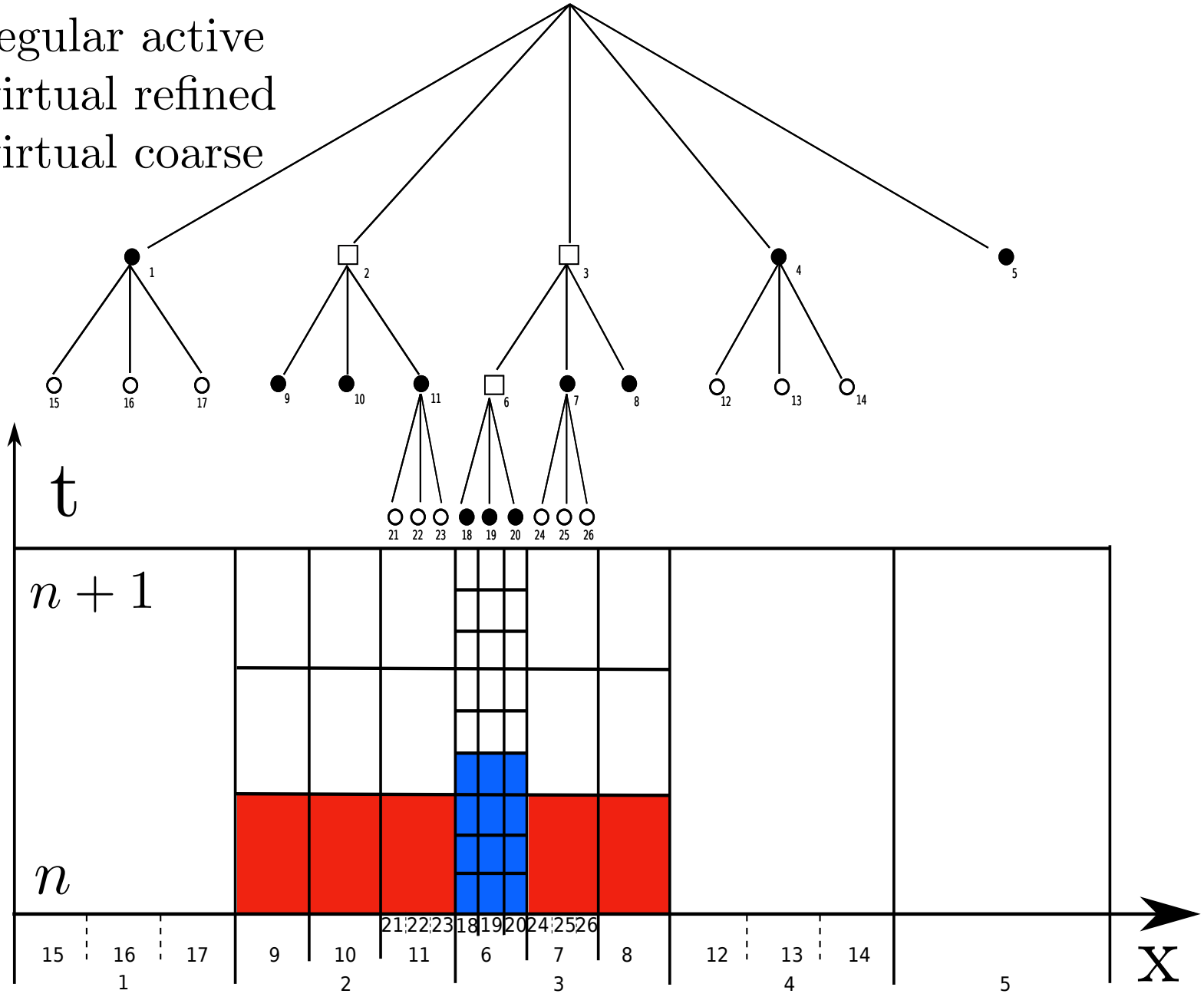
- regular active
- virtual refined
- virtual coarse



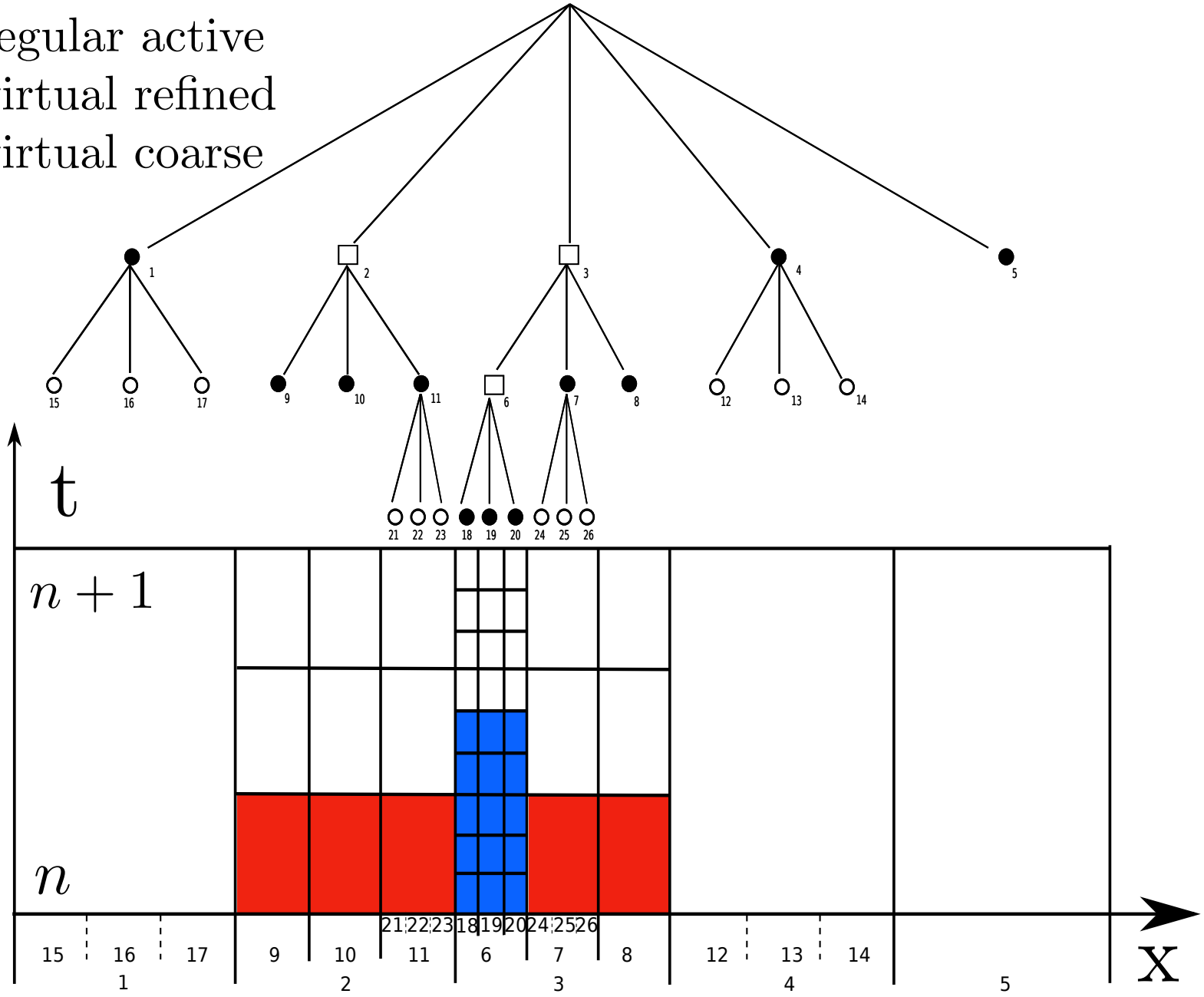
- regular active
- virtual refined
- virtual coarse



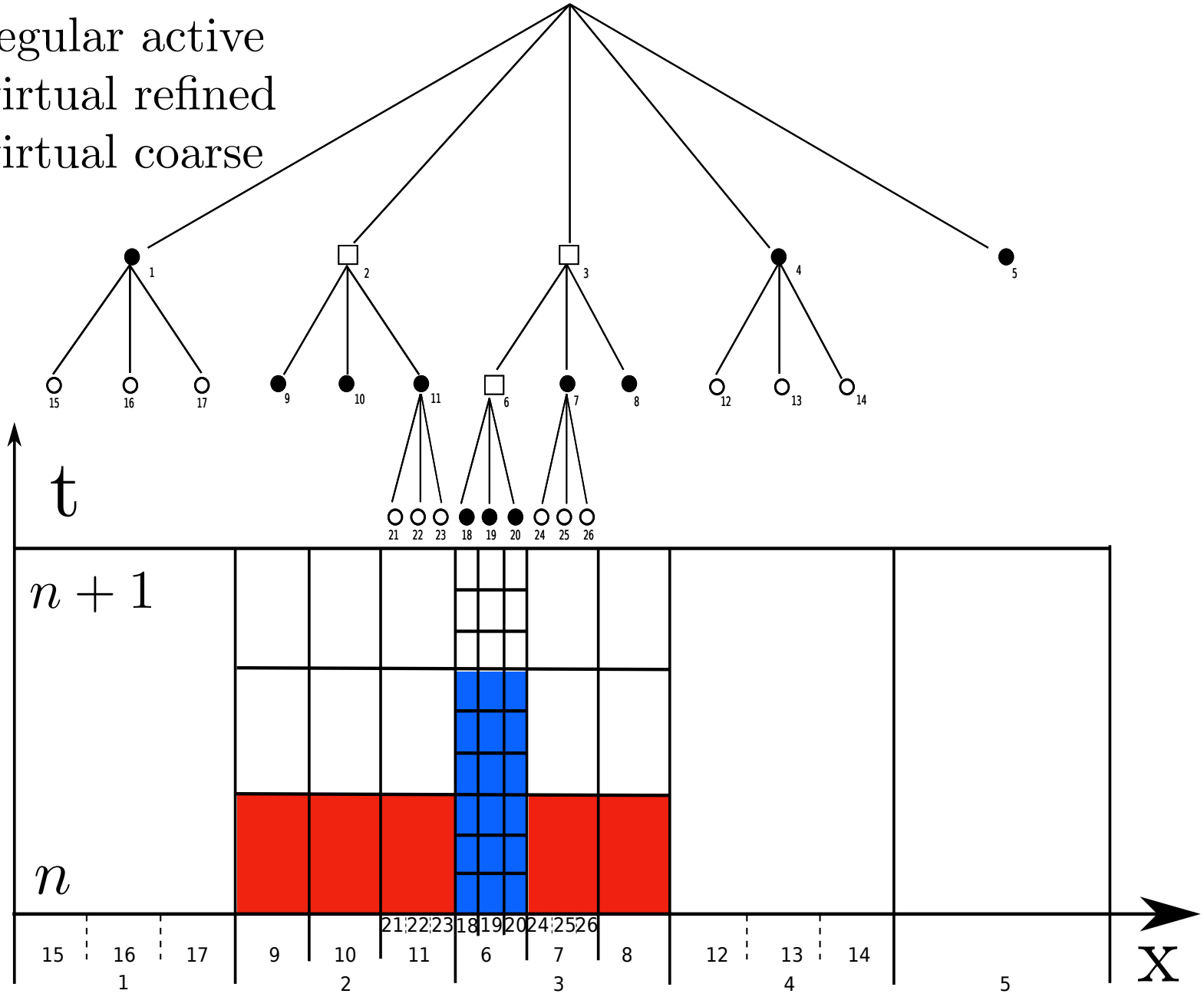
- regular active
- virtual refined
- virtual coarse



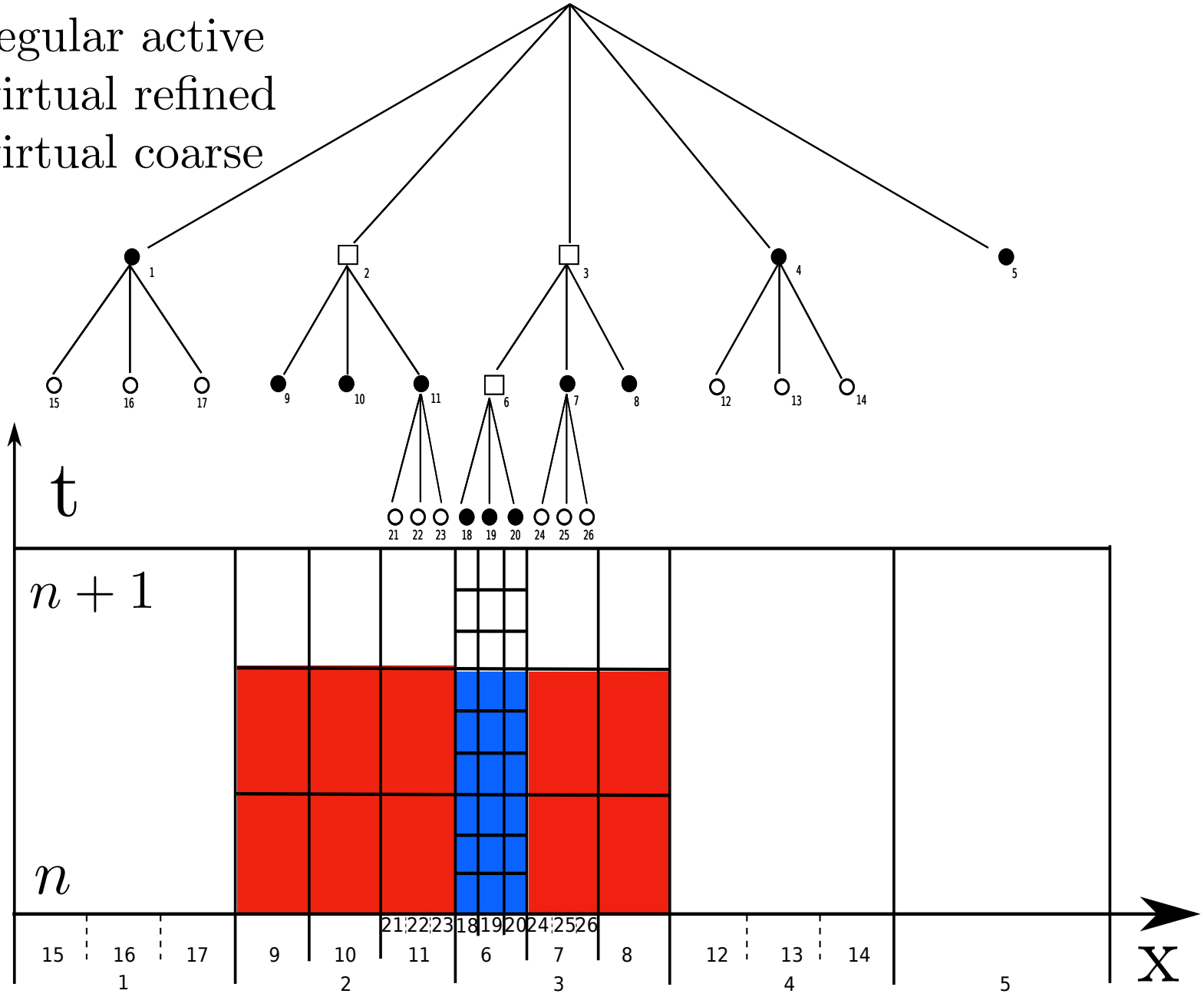
- regular active
- virtual refined
- virtual coarse



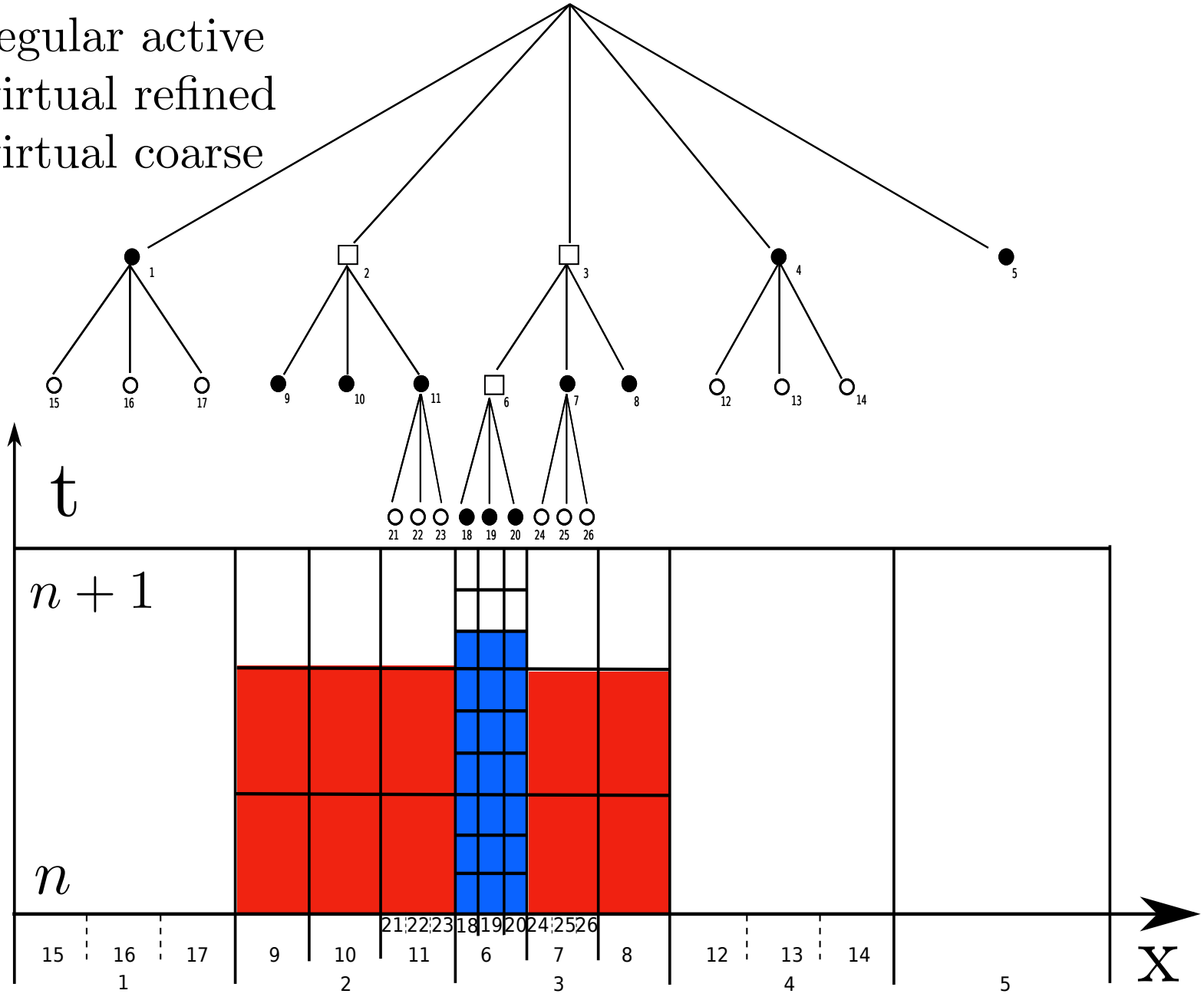
- regular active
- virtual refined
- virtual coarse



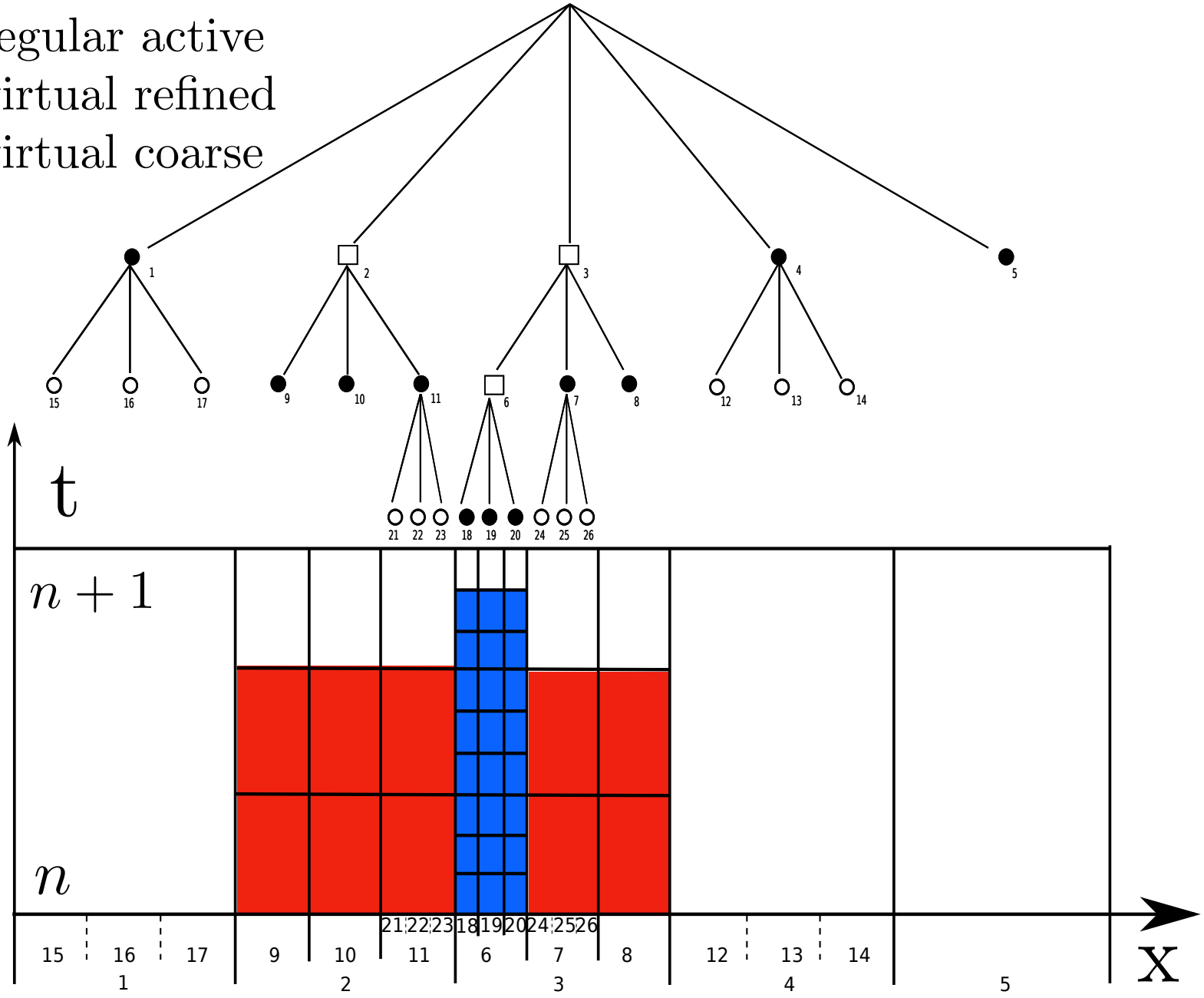
- regular active
- virtual refined
- virtual coarse



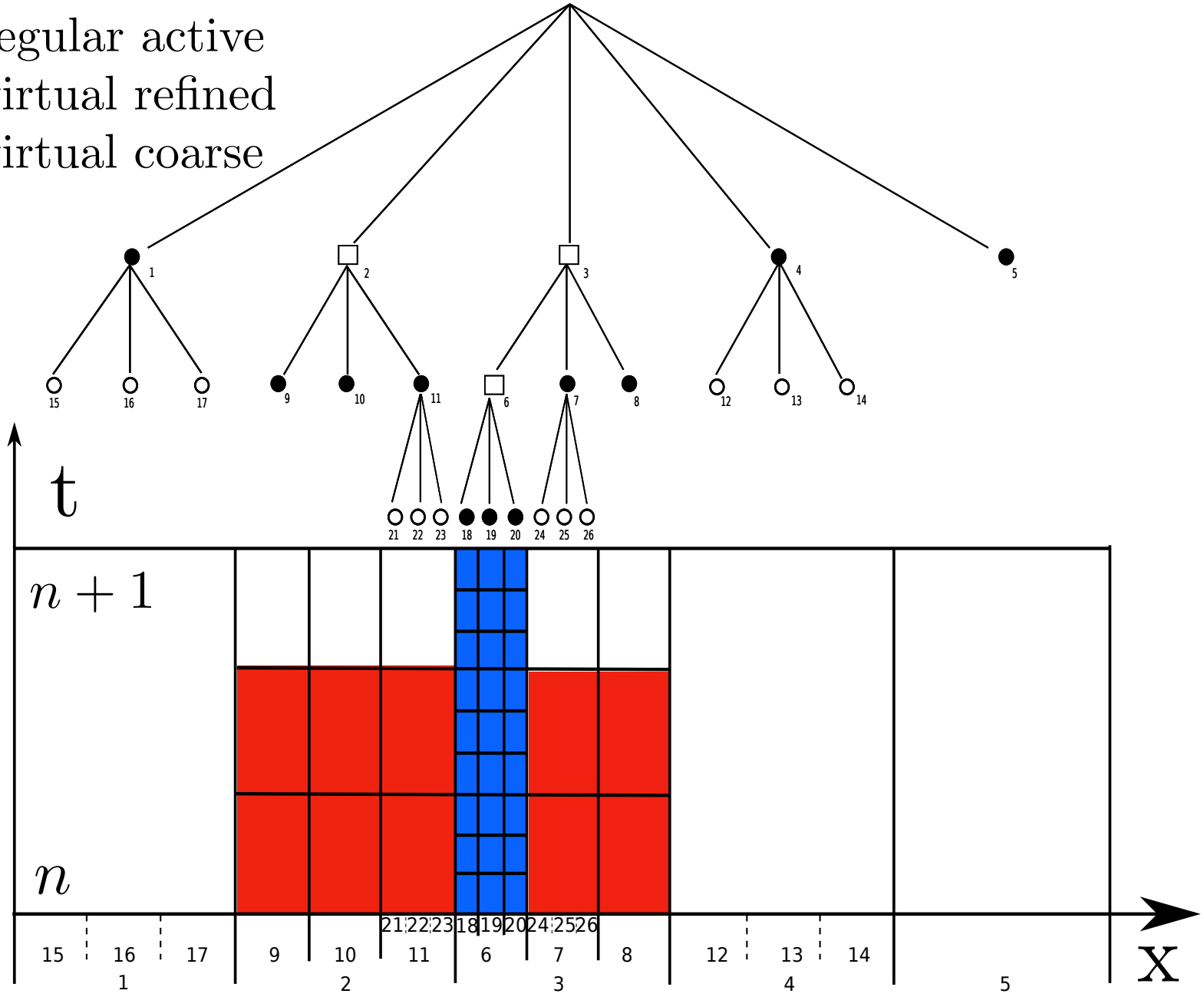
- regular active
- virtual refined
- virtual coarse



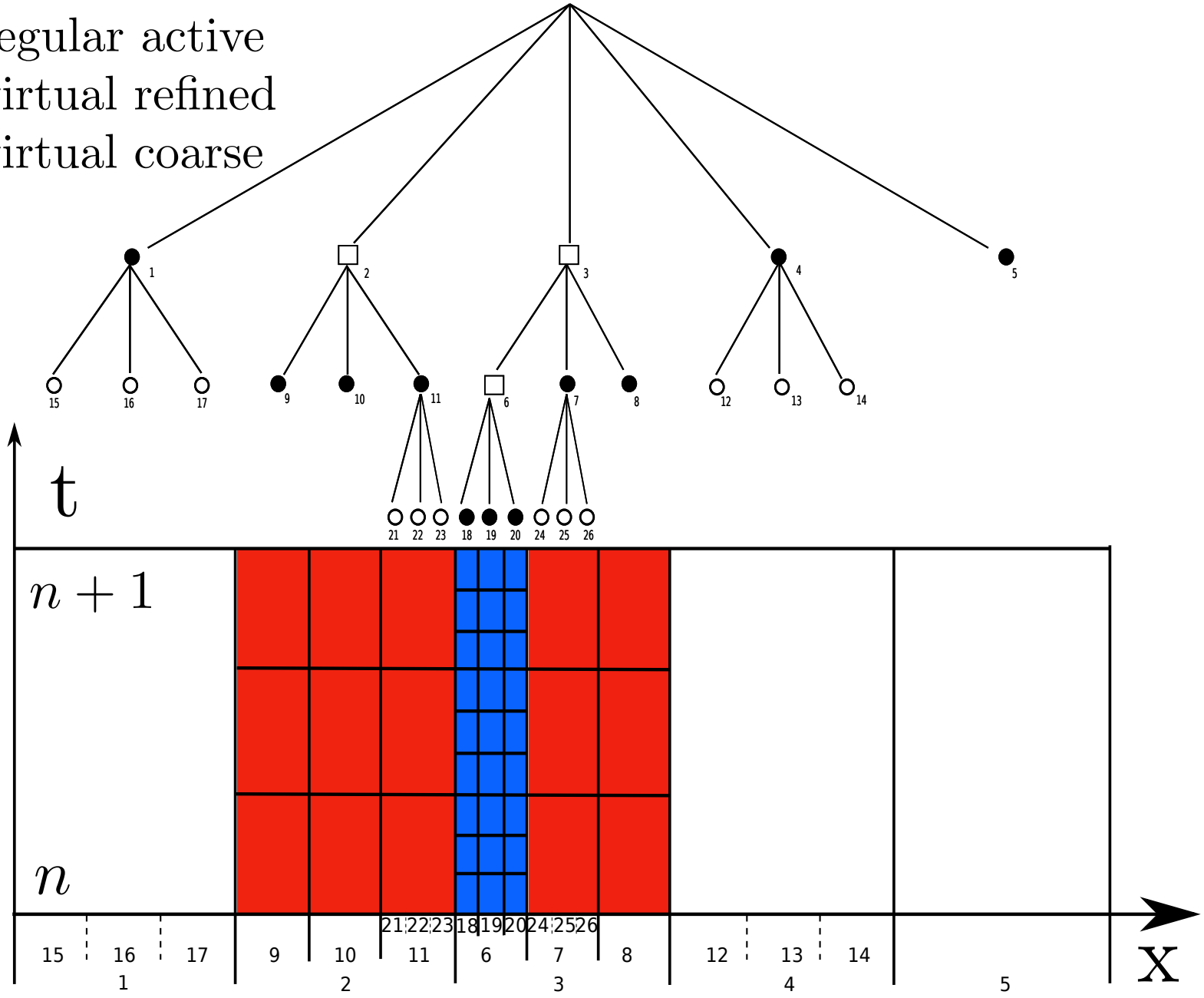
- regular active
- virtual refined
- virtual coarse

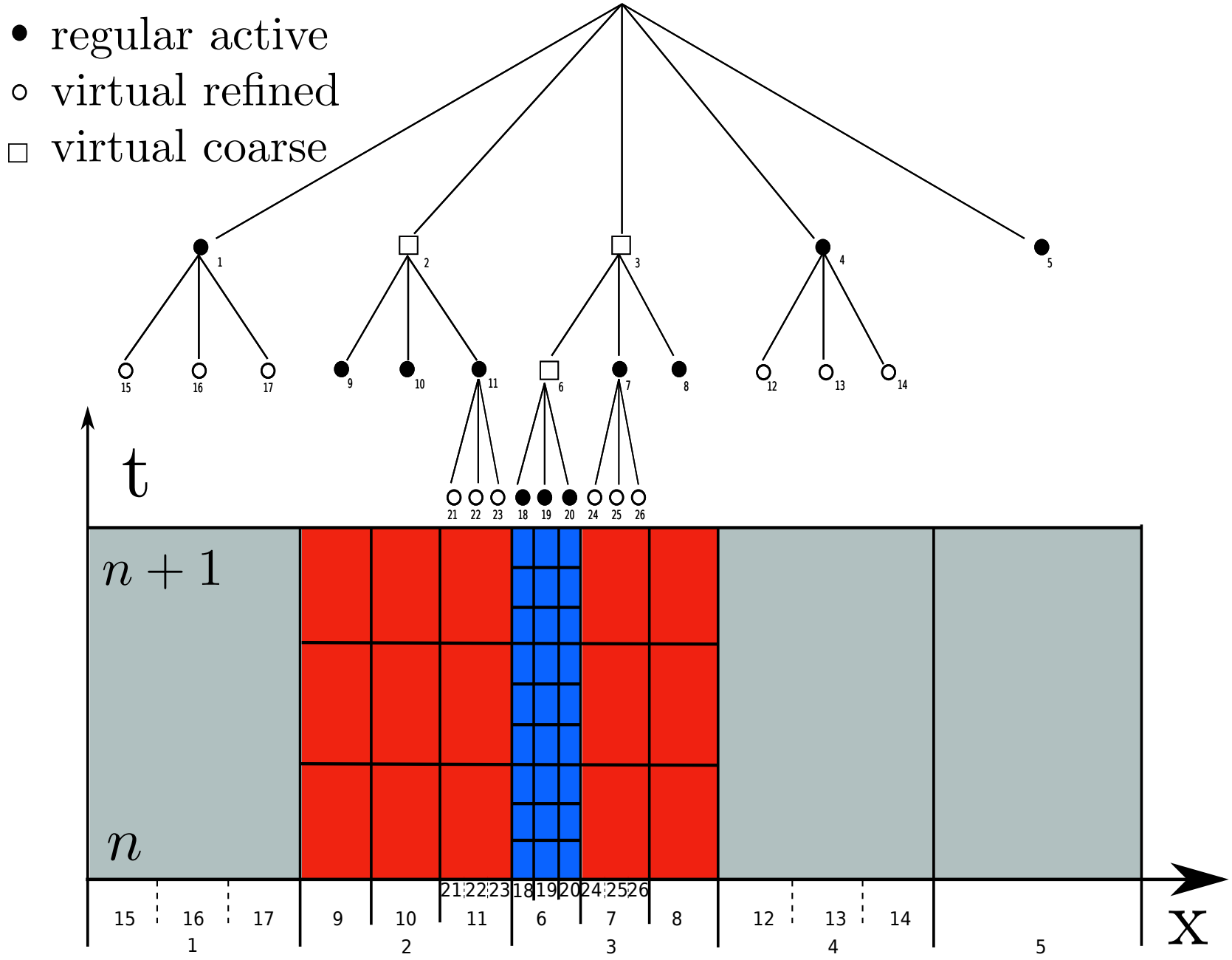


- regular active
- virtual refined
- virtual coarse



- regular active
- virtual refined
- virtual coarse




regular active
virtual refined
virtual coarse
1
2
3
4
5
15
16
17
9
10
11
6
7
8
12
13
14
21
22
23
18
19
20
24
25
26
t
$n+1$
n
X

Computation of numerical fluxes between adjacent cells on different levels of refinement

$$\mathbf{f}_{i+\frac{1}{2},jk} = \frac{1}{\Delta t_\ell}\frac{1}{\Delta y_\ell}\frac{1}{\Delta z_\ell}\sum_{ii=1}^{\mathfrak{r}}\sum_{jj=1}^{\mathfrak{r}}\sum_{kk=1}^{\mathfrak{r}}\int_{\mathcal{T}_{ii}}\int_{\mathcal{Y}_{jj}}\int_{\mathcal{Z}_{kk}}\tilde{\mathbf{f}}(\mathbf{q}_h^-,\mathbf{q}_h^+)\,dz\,dy\,dt, \qquad (6)$$

with the integration intervals defined as

$$\begin{aligned}
\mathcal{T}_{ii} &= [t_\ell^n + (ii-1)\Delta t_{\ell+1}; t_\ell^n + ii\Delta t_{\ell+1}],\\
\mathcal{Y}_{jj} &= [y_{j-\frac{1}{2}} + (jj-1)\Delta y_{\ell+1}; y_{j-\frac{1}{2}} + jj\Delta y_{\ell+1}],\\
\mathcal{Z}_{kk} &= [z_{k-\frac{1}{2}} + (kk-1)\Delta z_{\ell+1}; z_{j-\frac{1}{2}} + kk\Delta z_{\ell+1}]. \qquad (7)
\end{aligned}$$

Euler equations

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} + \frac{\partial \mathbf{h}}{\partial z} = \mathbf{0},$$

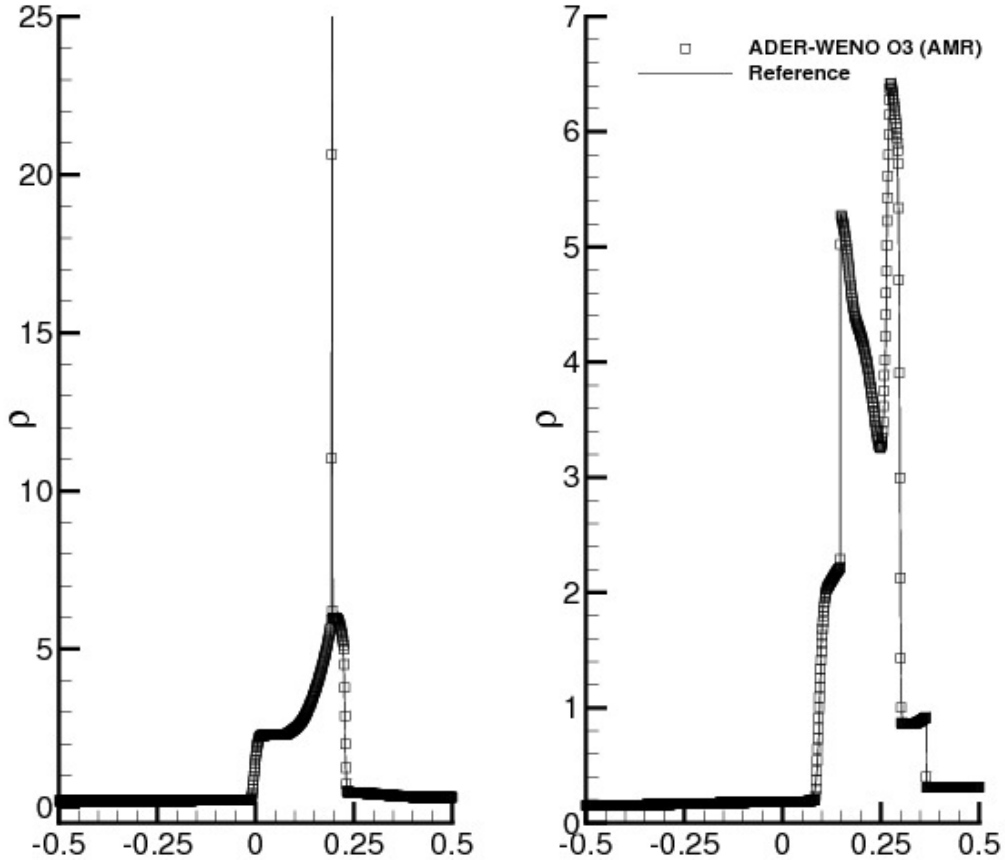
$$\mathbf{u} = \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho v_z \\ E \end{pmatrix}, \mathbf{f} = \begin{pmatrix} \rho v_x \\ \rho v_x^2 + p \\ \rho v_x v_y \\ \rho v_x v_z \\ v_x(E+p) \end{pmatrix}, \mathbf{g} = \begin{pmatrix} \rho v_y \\ \rho v_x v_y \\ \rho v_y^2 + p \\ \rho v_y v_z \\ v_y(E+p) \end{pmatrix},$$

$$\mathbf{h} = \begin{pmatrix} \rho v_z \\ \rho v_x v_z \\ \rho v_y v_z \\ \rho v_z^2 + p \\ v_z(E+p) \end{pmatrix},$$

where v_x, v_y and v_z are the velocity components, p the pressure, ρ the mass density, $E = p/(\gamma - 1) + \rho(v_x^2 + v_y^2 + v_z^2)/2$ the total energy density, γ the adiabatic index.

Interacting blast waves

$$(\rho, v_x, p) = \begin{cases} (1.0, 0.0, 10^3) & \text{if} \quad -0.5 < x < -0.4\,, \\ (1.0, 0.0, 10^{-2}) & \text{if} \quad -0.4 < x < 0.4\,, \\ (1.0, 0.0, 10^2) & \text{if} \quad 0.4 < x < 0.5\,. \end{cases}$$

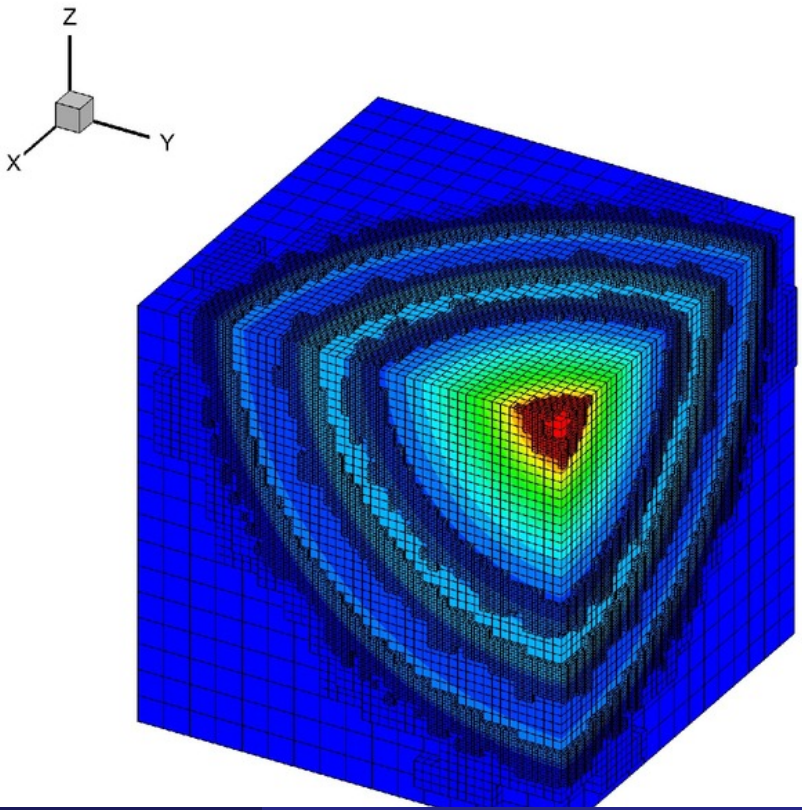


- Reflecting boundary conditions
- Two levels of refinement used

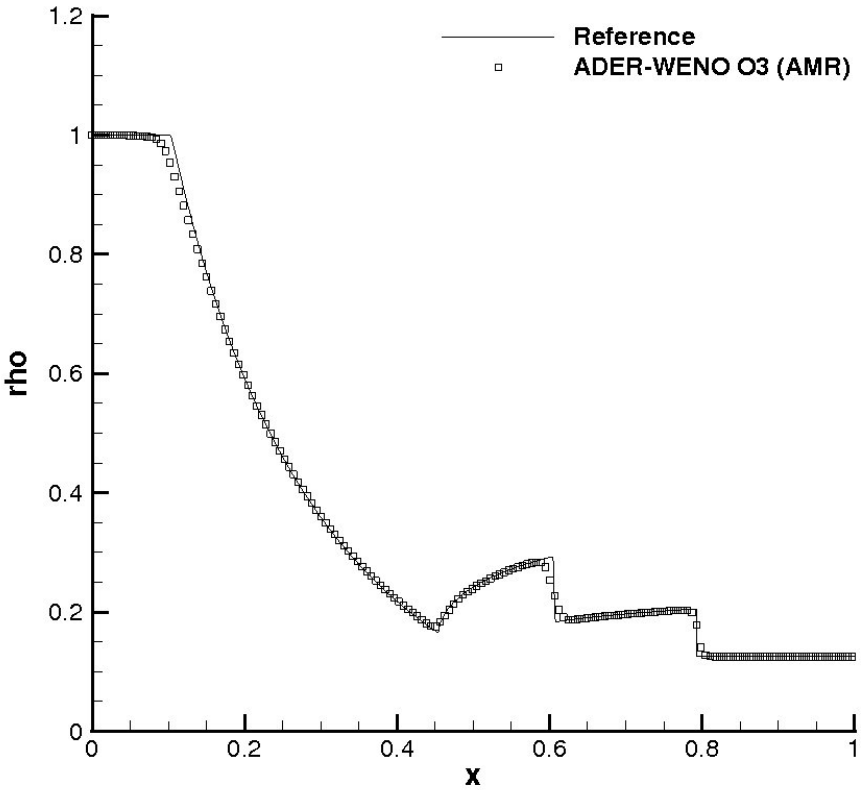
Explosion problem in 3D

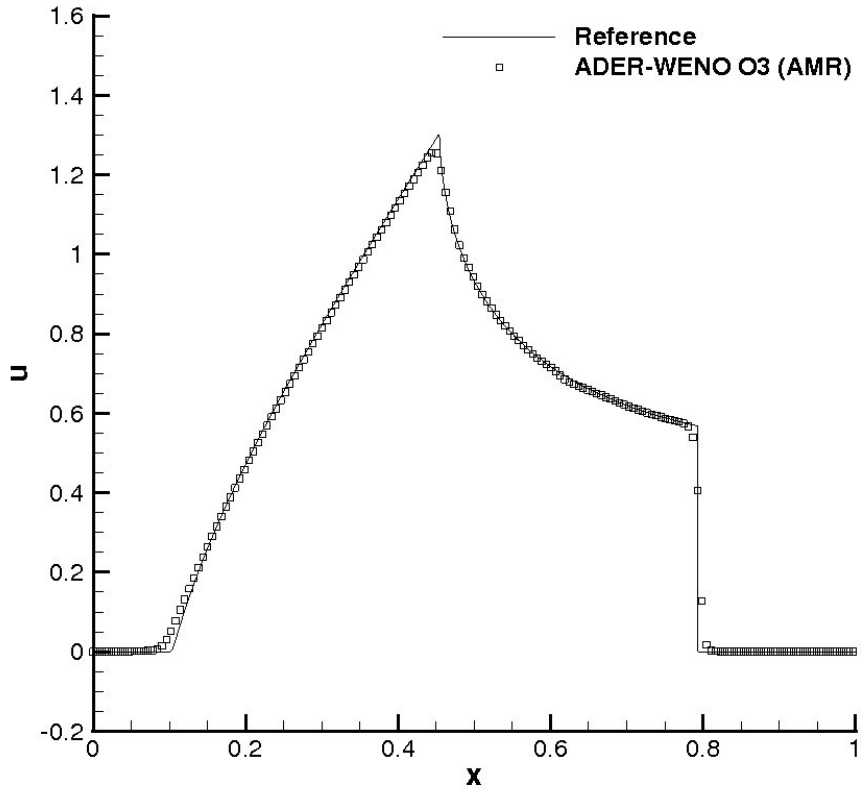
$$\mathbf{u}(\mathbf{x},0) = \begin{cases} \mathbf{u}_i & \text{if} \quad r \leq R, \\ \mathbf{u}_o & \text{if} \quad r > R. \end{cases}$$

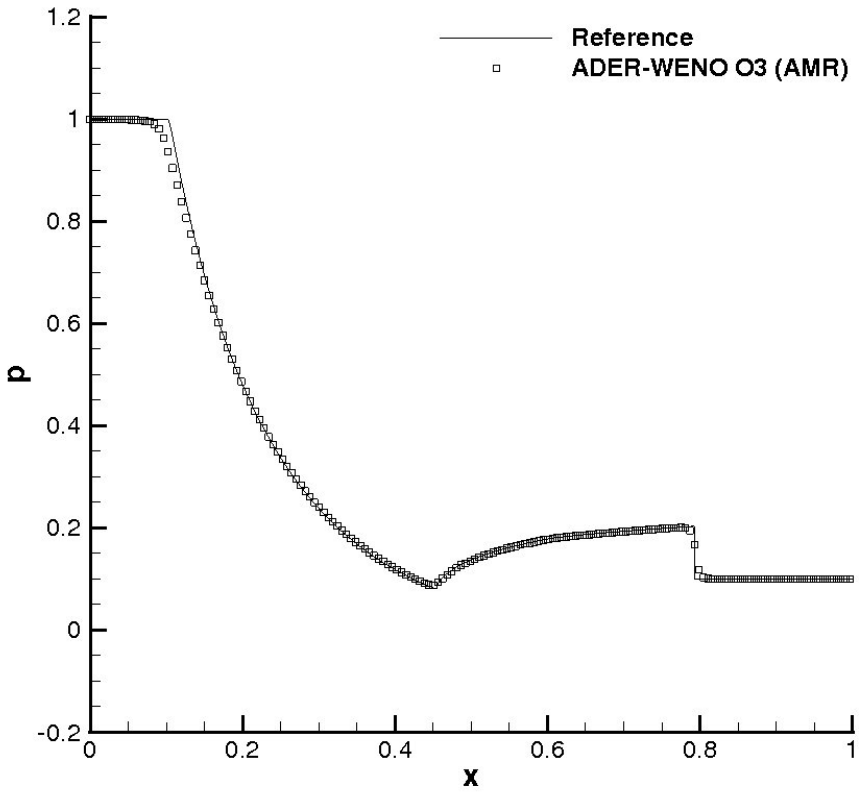
	ρ	p	v_x	v_y	v_z	t_e
Inner	1.0	1.0	0.0	0.0	0.0	0.25
Outer	0.125	0.1	0.0	0.0	0.0	

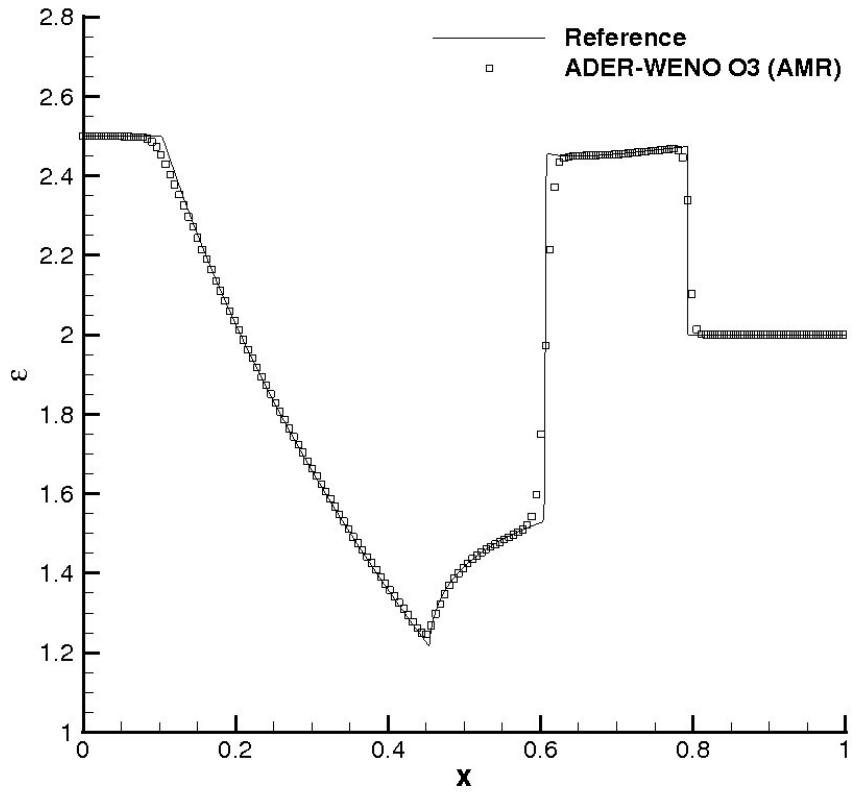


- a level zero mesh with $34x34x34$ cells.
- $\ell_{max} = 2$, $\mathfrak{r} = 3$, corresponding to a fine grid resolution of $306^3 = 28,652,616$ cells.
- The final grid has 9079984 cells.


Reference
ADER-WENO O3 (AMR)
rho
x
0
0.2
0.4
0.6
0.8
1
1.2

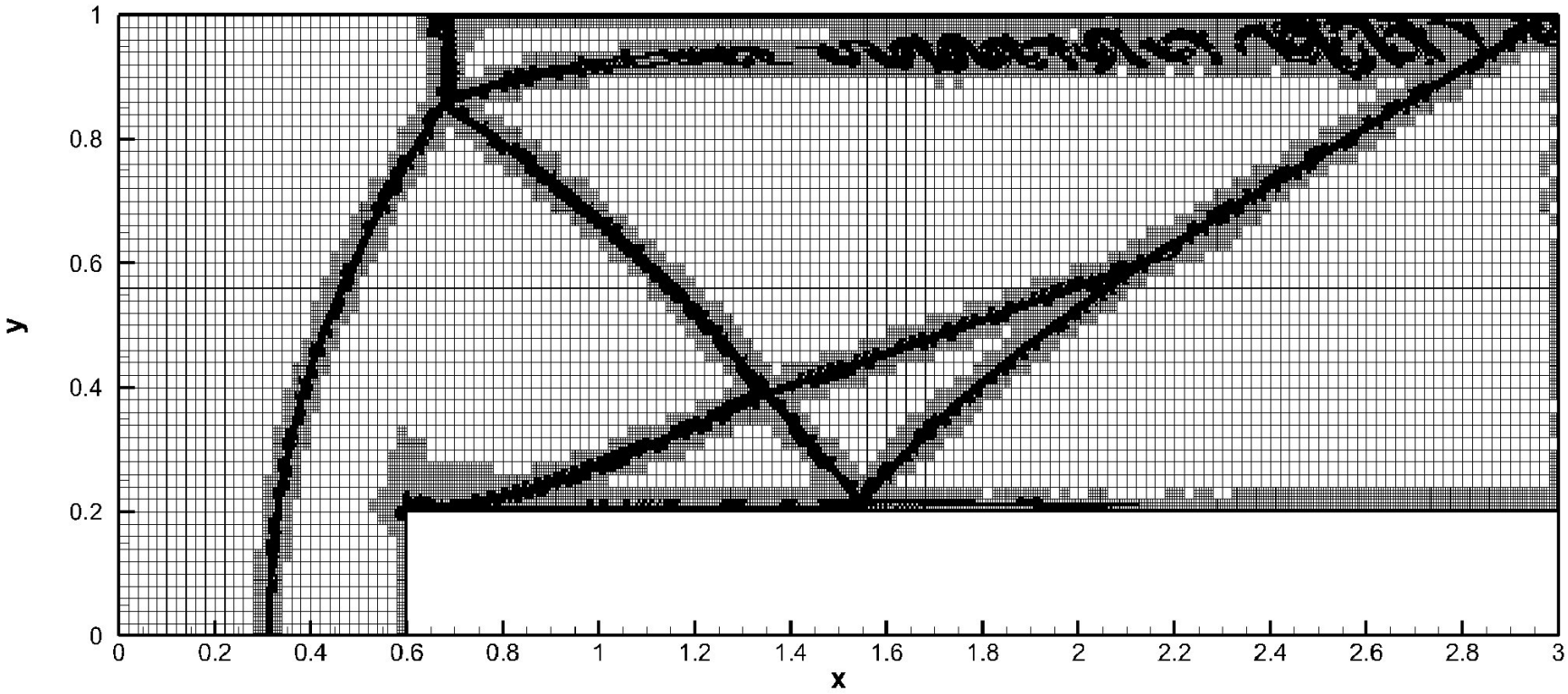

Reference
ADER-WENO O3 (AMR)
u
x
-0.2
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6


Reference
ADER-WENO O3 (AMR)
ρ
x
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
0.2
0.4
0.6
0.8
1

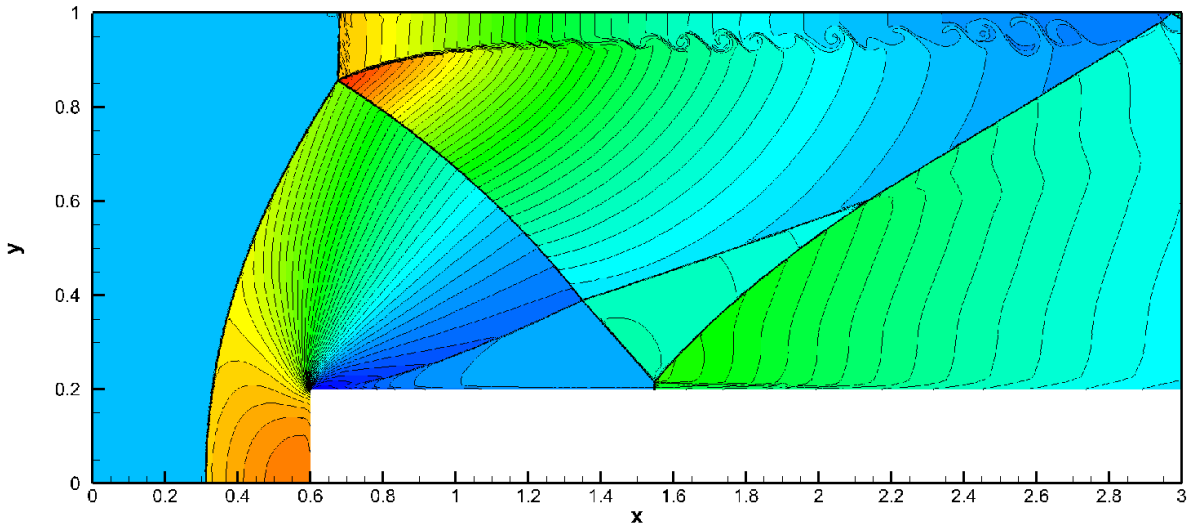

Reference
ADER-WENO O3 (AMR)
ε
x
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
1.2
1
0
0.2
0.4
0.6
0.8
1

Forward facing step problem

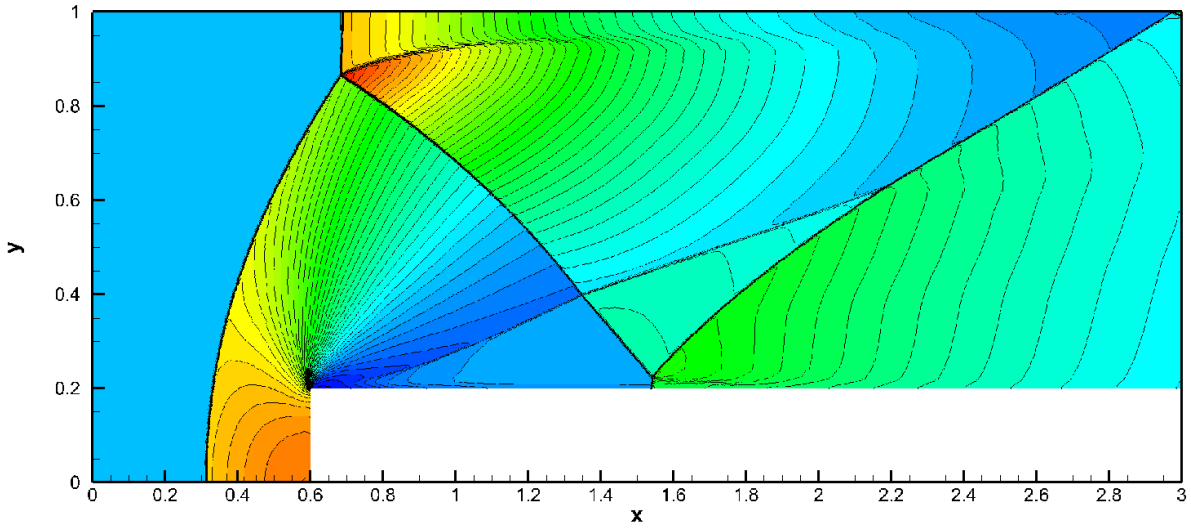
The initial condition is a uniform flow at Mach number $M = 3$ moving to the right: $\rho(x, y, 0) = 1$, $p(x, y, 0) = 1/\gamma$, $v_x(x, y, 0) = 3$, $v_y = v_z = 0$, $\gamma = 1.4$. Mesh on the coarsest level with 150×50 cells, $\mathfrak{r} = 4$, $\ell_{\max} = 2$,



Third order



Second order



Co-rotating vortex pair

A typical multi-scale problem from aeroacoustics, consisting of two isentropic vortices with characteristic size r_c (vortex core radius) that rotate around each other, generating sound waves with a $\lambda/r_c \simeq 10^2$. The complex potential w of the rotating vortex pair is given by

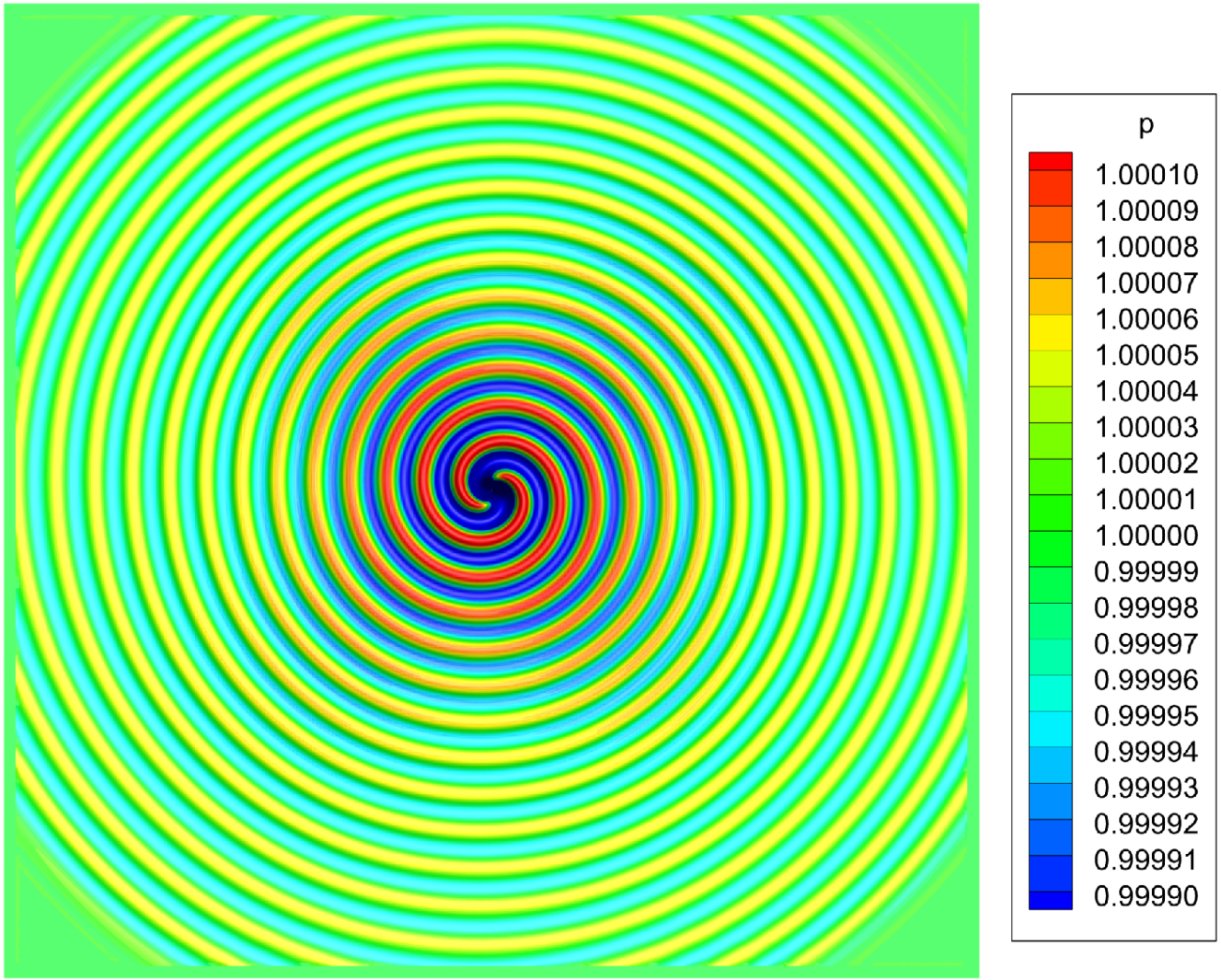
$$w(z,t) = \frac{\Gamma}{2\pi i} \ln\left(z^2 - b^2\right) \Longrightarrow \frac{\partial w}{\partial z} = \frac{\Gamma}{\pi i} \frac{z}{z^2 - b^2} = v_x - i v_y. \tag{8}$$

with $z = x + iy$, $b = r_0 e^{i\omega t}$, $\omega = \Gamma/(4\pi r_0^2)$ and the rotation Mach number is $M = \Gamma/(4\pi r_0 c_0)$.
The far field sound pressure is

$$p' = \frac{\rho_0 \Gamma^4}{64\pi^3 r_0^4 c_0^2} \left(\mathrm{J}_2(kr)\sin\left(2(\omega t - \theta)\right) - \mathrm{Y}_2(kr)\cos\left(2(\omega t - \theta)\right)\right), \tag{9}$$

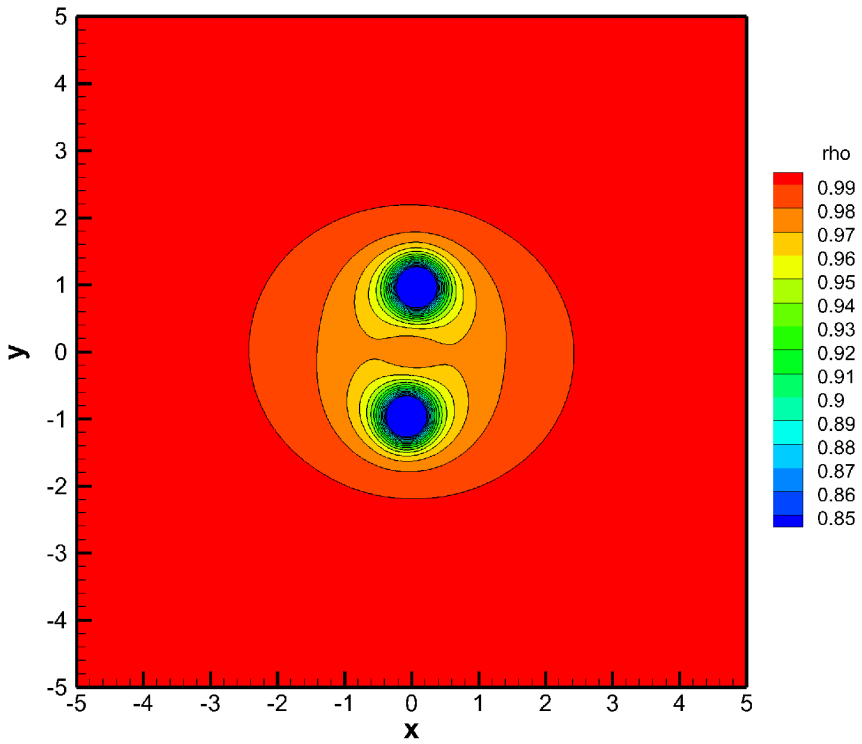
with $k = 2\omega/c_0$.
For the **numerical simulations**, the two-dimensional computational domain of this problem is chosen as $\Omega = [-500; 500] \times [-500; 500]$. We used a fourth order ADER-WENO scheme using a level zero grid of 250×250 elements, $\mathfrak{r} = 4$, $\ell_{\max} = 3$.

The pressure field at $t = 500$

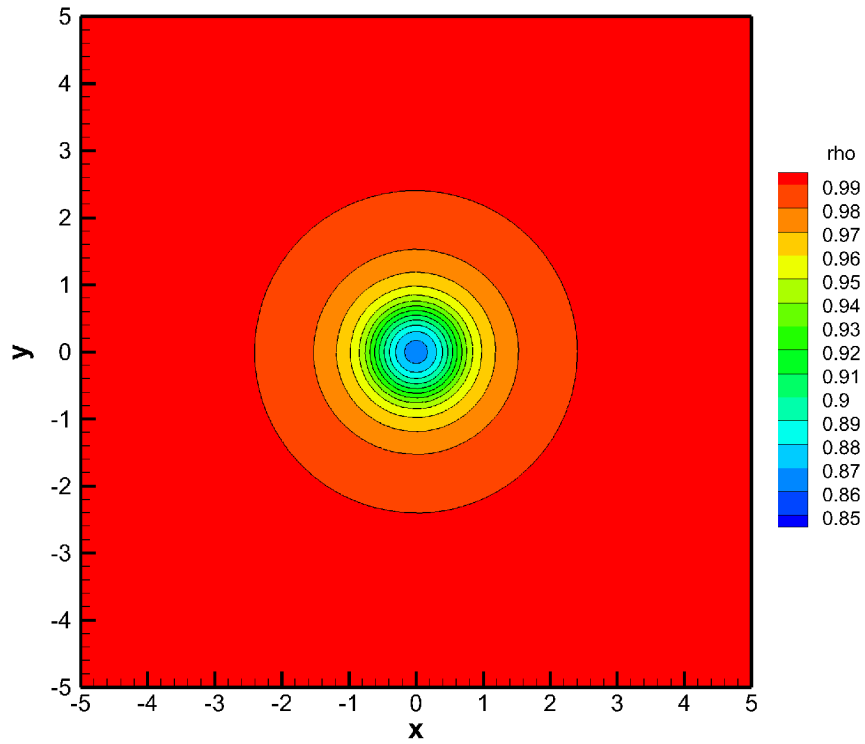


How important is high order?

Density contours of the co-rotating vortex pair at time $t = 500$

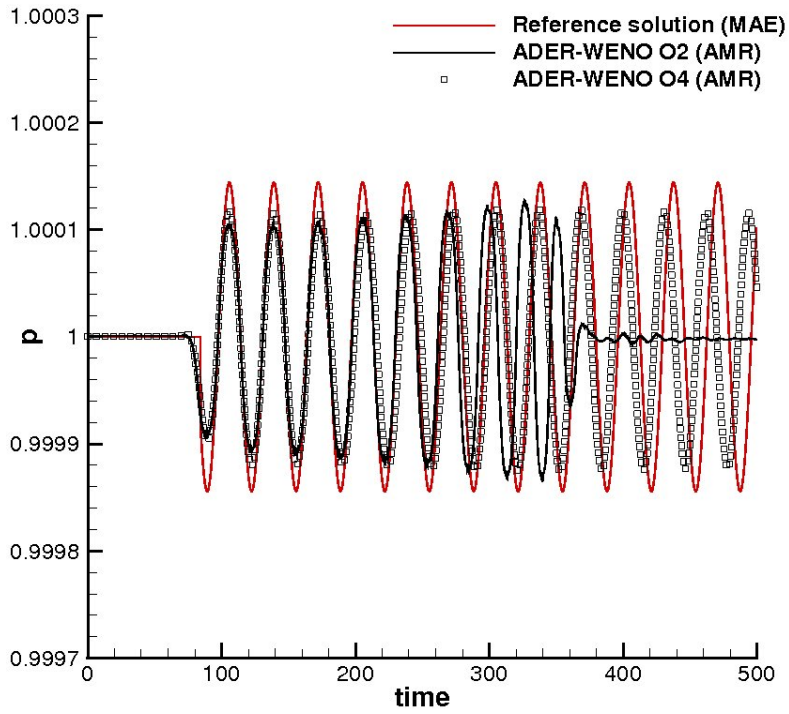


Fourth Order

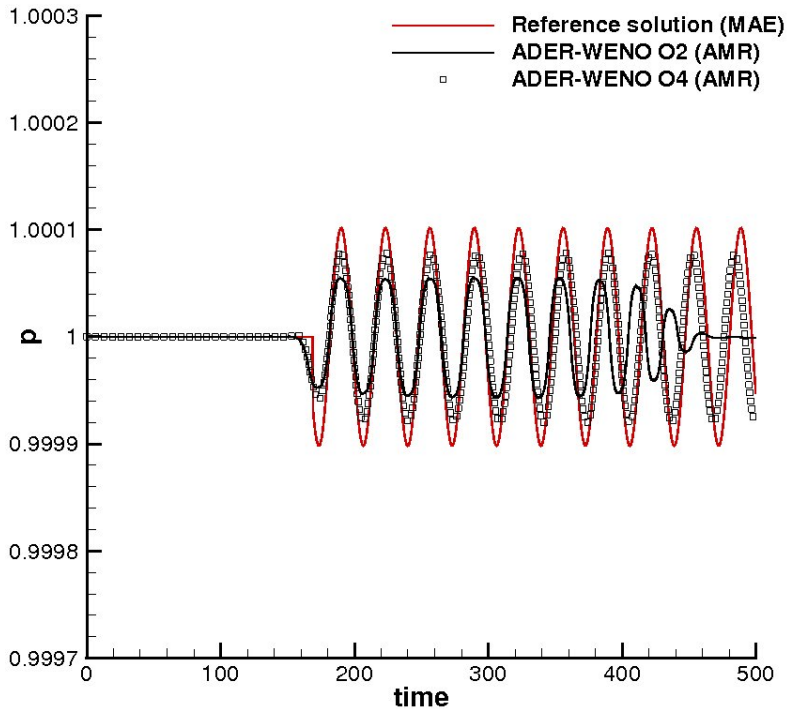


Second Order

How important is high order?



Measured at $x = 100$



Measured at $x = 200$

MHD equations

The equations of Ideal Magnetohydrodynamics are not hyperbolic

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{v}) = 0$$
$$\frac{\partial}{\partial t}(\rho \vec{v}) + \vec{\nabla} \cdot \left[\rho \vec{v} \vec{v} + (p + \frac{1}{8\pi} \vec{B}^2) \mathbf{I} - \frac{1}{4\pi} \vec{B} \vec{B} \right] = 0$$
$$\frac{\partial}{\partial t}(\frac{1}{2} \rho v^2 + \rho \epsilon + \frac{1}{8\pi} \vec{B}^2) + \vec{\nabla} \cdot \left[(\frac{1}{2} \rho v^2 + \rho \epsilon + p + \frac{1}{8\pi} \vec{B}^2) \vec{v} - \vec{v} \frac{1}{4\pi} \vec{B} \vec{B} \right] = 0$$
$$\frac{\partial \vec{B}}{\partial t} + \vec{\nabla} \cdot (\vec{v} \vec{B} - \vec{B} \vec{v}) = 0$$
$$\vec{\nabla} \cdot \vec{B} = 0$$

We use hyperbolic divergence cleaning after introducing an additional scalar Ψ

$$\frac{\partial \mathbf{u}}{\partial t} + \vec{\nabla} \cdot \mathbf{F} = \mathbf{0}, \tag{10}$$

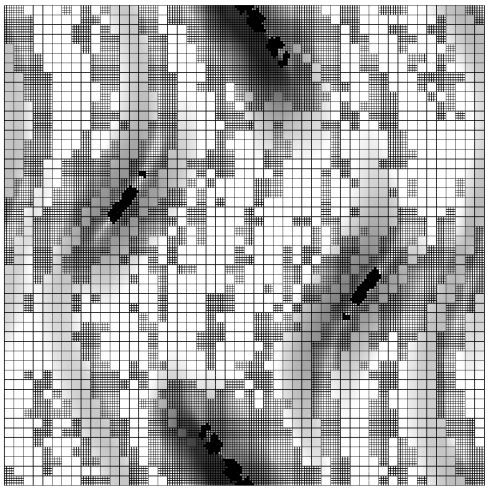
$$\mathbf{u} = \begin{pmatrix} \rho \\ \rho \vec{v} \\ E \\ \vec{B} \\ \psi \end{pmatrix}, \mathbf{F} = \begin{pmatrix} \rho \vec{v} \\ \rho \vec{v} \vec{v} + (p + \frac{1}{8\pi} \vec{B}^2) \mathbf{I} - \frac{1}{4\pi} \vec{B} \vec{B} \\ \vec{v}(E + p + \frac{1}{8\pi} \vec{B}^2) - \frac{1}{4\pi} \vec{B}(\vec{v} \cdot \vec{B}) \\ \vec{v} \vec{B} - \vec{B} \vec{v} + \Psi \mathbf{I}, \\ c_h^2 \vec{B} \end{pmatrix}. \tag{11}$$

Orszag-Tang vortex system

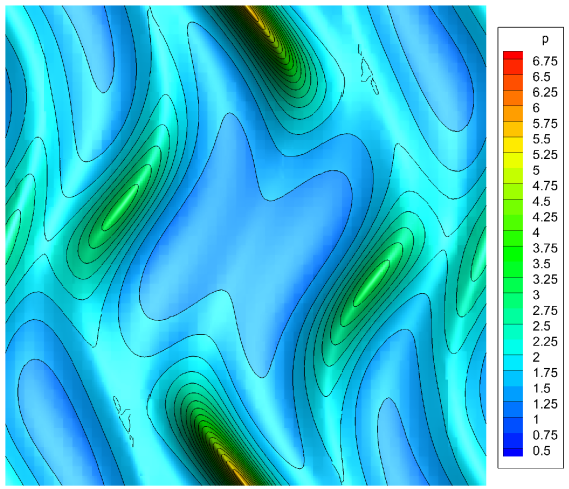
Initial conditions:

$$(\rho, u, v, p, B_x, B_y) = \left(\gamma^2, -\sin(y), \sin(x), \gamma, -\sqrt{4\pi}\sin(y), \sqrt{4\pi}\sin(2x)\right),$$

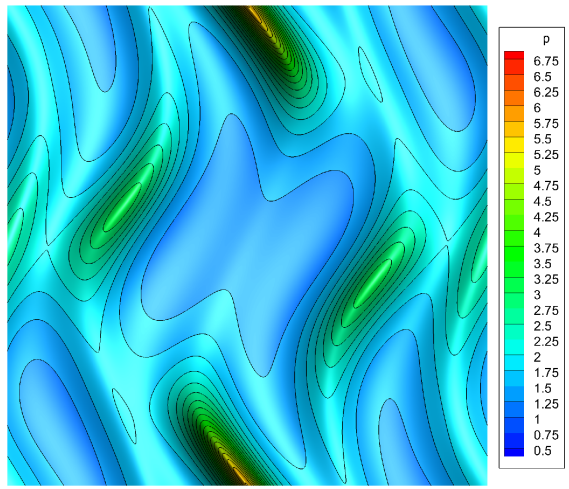
with $w = B_z = 0$, $\gamma = 5/3$, $\Omega = [0; 2\pi]^2$, $c_h = 2$.



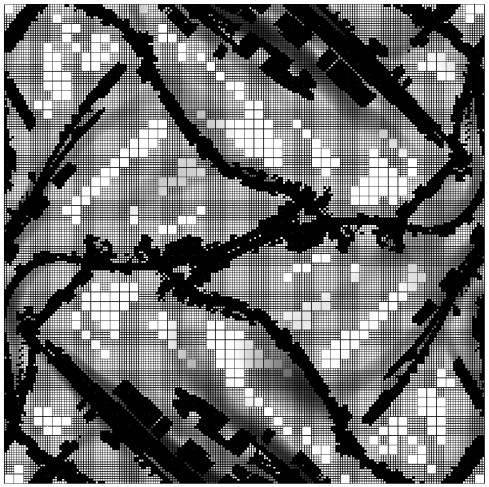
AMR grid

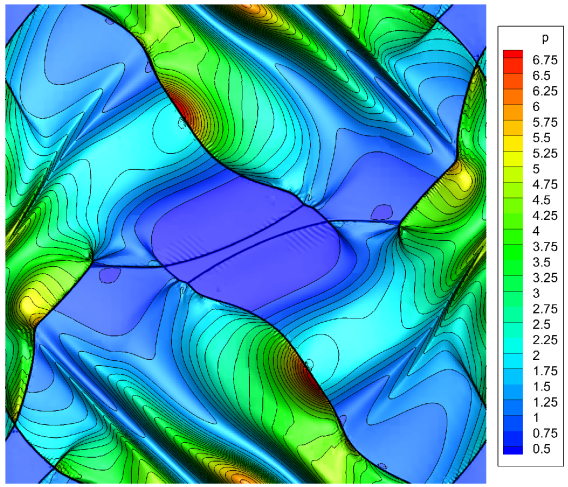


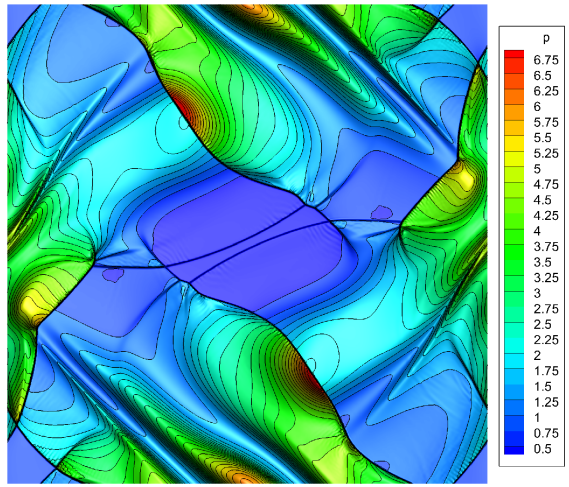
AMR O3

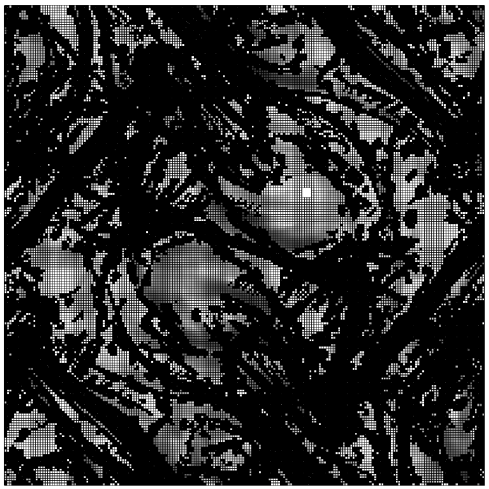


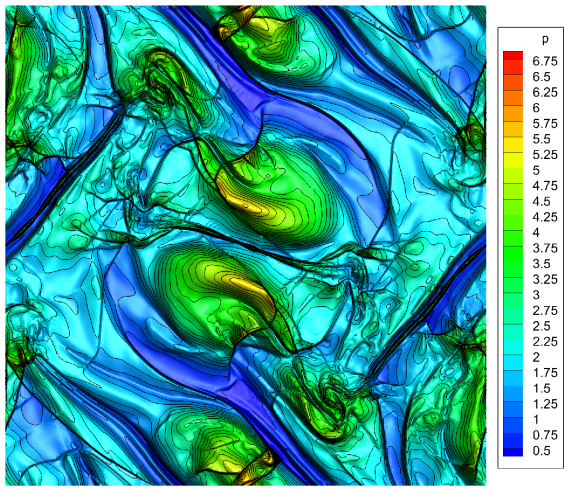
Uniform-finest

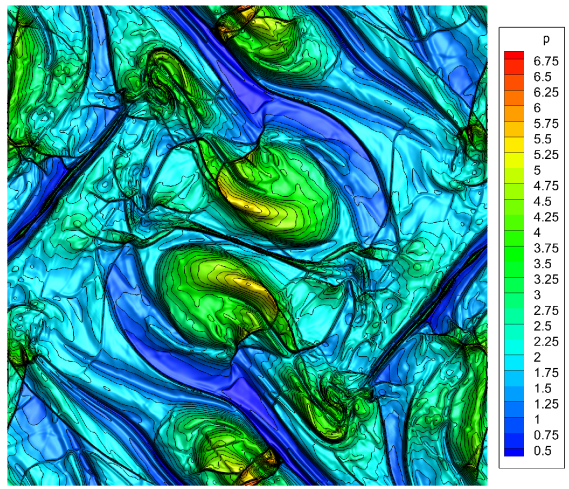












AMR grid **AMR O3** **Uniform-finest**

Is all this efficient?

- AMR overhead (on single CPU): Normalized average CPU time per real element update. ($\mathfrak{r} = 4, \ell = 2$).

Scheme order	Uniform grid	AMR grid	Total AMR overhead
$\mathcal{O}2$	1.00	1.15	15 %
$\mathcal{O}3$	3.18	3.82	20 %
$\mathcal{O}4$	8.64	10.82	25 %

- Memory and CPU time comparison (from Orszag-Tang problem)

	AMR	Uniform	ratio
Cells	454525	640000	1.41
CPU	0.547	1.0	1.83

- Memory and CPU time comparison (from MHD rotor problem)

	AMR	Uniform	ratio
Elements	179680	921600	5.13
CPU time	0.140	1.0	7.14

Conclusions

Conclusions

- We have presented the first ADER-WENO finite volume scheme on AMR grids

Conclusions

- We have presented the first ADER-WENO finite volume scheme on AMR grids
- Compared to Runge–Kutta time stepping, the use of a high order one–step scheme in time reduces the number of nonlinear WENO reconstructions and the number of necessary MPI communications.

Conclusions

- We have presented the first ADER-WENO finite volume scheme on AMR grids
- Compared to Runge–Kutta time stepping, the use of a high order one–step scheme in time reduces the number of nonlinear WENO reconstructions and the number of necessary MPI communications.
- Even in the AMR context the use of higher order schemes is beneficial, in particular when small scale turbulent structures

Conclusions

- We have presented the first ADER-WENO finite volume scheme on AMR grids
- Compared to Runge–Kutta time stepping, the use of a high order one–step scheme in time reduces the number of nonlinear WENO reconstructions and the number of necessary MPI communications.
- Even in the AMR context the use of higher order schemes is beneficial, in particular when small scale turbulent structures
- A new tool for several astropysical applications is now available

Conclusions

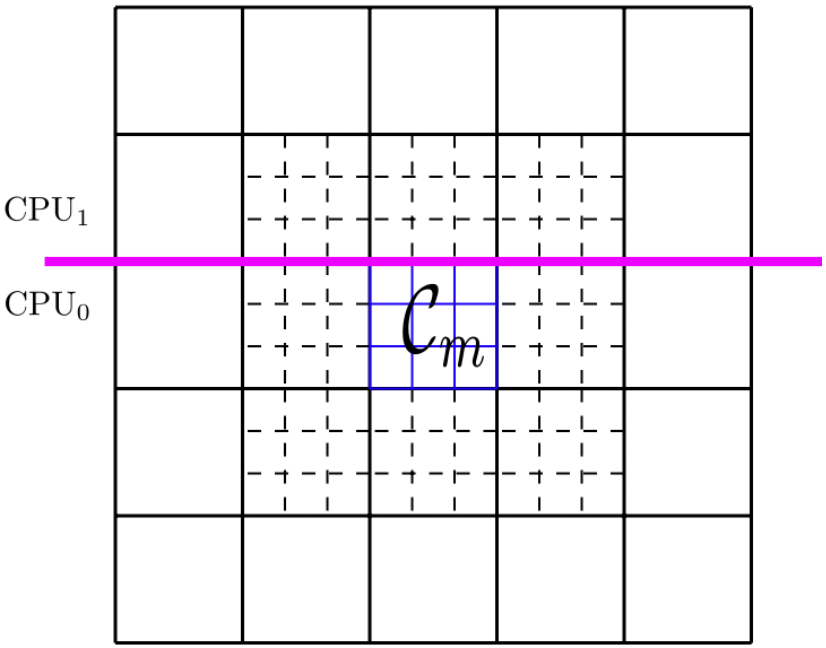
- We have presented the first ADER-WENO finite volume scheme on AMR grids
- Compared to Runge–Kutta time stepping, the use of a high order one–step scheme in time reduces the number of nonlinear WENO reconstructions and the number of necessary MPI communications.
- Even in the AMR context the use of higher order schemes is beneficial, in particular when small scale turbulent structures
- A new tool for several astropysical applications is now available

Thank you!

olindo.zanotti@unitn.it

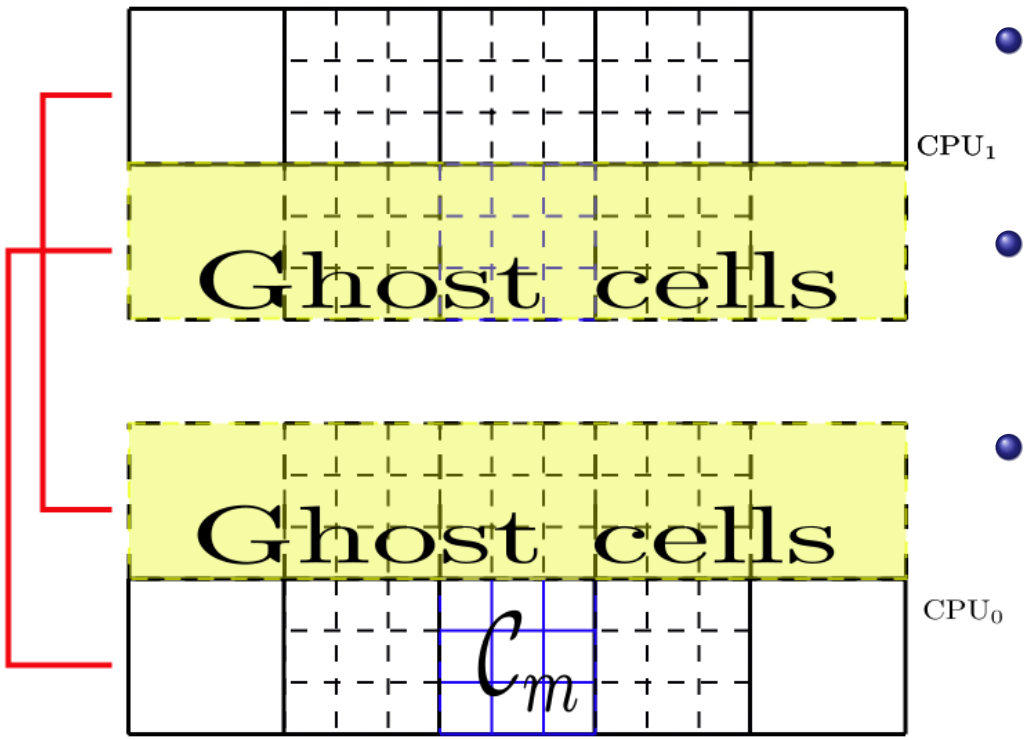
AMR Parallelization

AMR poses additional challenges to the parallelization task, which emerge when a cell is refined at the MPI border between two processors



- Each processor must know which cells must be virtually refined or activated;
- CPU_0 must inform CPU_1 that a number of real cells in the domain of CPU_1 need virtual refinement;
- We found convenient to introduce **MPI-ghost zones**, that are a copy of the true cells, managed by the adjacent processor.

AMR Parallelization



- CPU_0 must inform CPU_1 that a number of real cells in the domain of CPU_1 need virtual refinement;
- CPU_1 checks whether such cells have already received an instruction of virtual refinement internal to CPU_1;
- We use *exchange lists* to perform the link between the true cells of CPU_0 and those belonging to the MPI-ghost zone of CPU_1.

Computation of the numerical flux at an interface between elements of level ℓ and $\ell+1$.

$$\mathbf{f}_{i+\frac{1}{2},jk} = \frac{1}{\Delta t_\ell}\frac{1}{\Delta y_\ell}\frac{1}{\Delta z_\ell}\sum_{ii=1}^{\mathfrak{r}}\sum_{jj=1}^{\mathfrak{r}}\sum_{kk=1}^{\mathfrak{r}}\int_{\mathcal{T}_{ii}}\int_{\mathcal{Y}_{jj}}\int_{\mathcal{Z}_{kk}}\tilde{\mathbf{f}}(\mathbf{q}_h^-,\mathbf{q}_h^+)\,dz\,dy\,dt, \qquad (12)$$

with the integration intervals above defined as

$$\begin{aligned}
\mathcal{T}_{ii} &= [t_\ell^n + (ii-1)\Delta t_{\ell+1}; t_\ell^n + ii\Delta t_{\ell+1}],\\
\mathcal{Y}_{jj} &= [y_{j-\frac{1}{2}} + (jj-1)\Delta y_{\ell+1}; y_{j-\frac{1}{2}} + jj\Delta y_{\ell+1}],\\
\mathcal{Z}_{kk} &= [z_{k-\frac{1}{2}} + (kk-1)\Delta z_{\ell+1}; z_{j-\frac{1}{2}} + kk\Delta z_{\ell+1}]. \qquad (13)
\end{aligned}$$

Tabella : Numerical convergence results for the isentropic vortex test using the third and fourth order version of the one–step ADER-WENO finite volume scheme presented in this article. The error norms refer to the variable ρ (density) at the final time $t_f = 10$. The asterisk * refers to a uniform grid.

$\ell_{\max} = 2$					
$N_G \times N_G$	ϵ_{L_2}	$\mathcal{O}(L_2)$	$N_G \times N_G$	ϵ_{L_2}	$\mathcal{O}(L_2)$
		$\mathcal{O}3$			$\mathcal{O}4$
12×12*	5.0131E-01		10×10*	5.1496E-01	
24×24	1.5223E-02	5.04	15×15	3.2990E-02	6.78
36×36	5.6974E-03	4.08	21×21	1.2157E-02	5.05
48×48	2.3935E-03	3.86	28×28	4.5922E-03	4.58
72×72	7.5147E-04	3.63	42×42	1.0334E-03	4.33
108×108	5.4038E-04	3.11	63×63	2.4593E-04	4.15