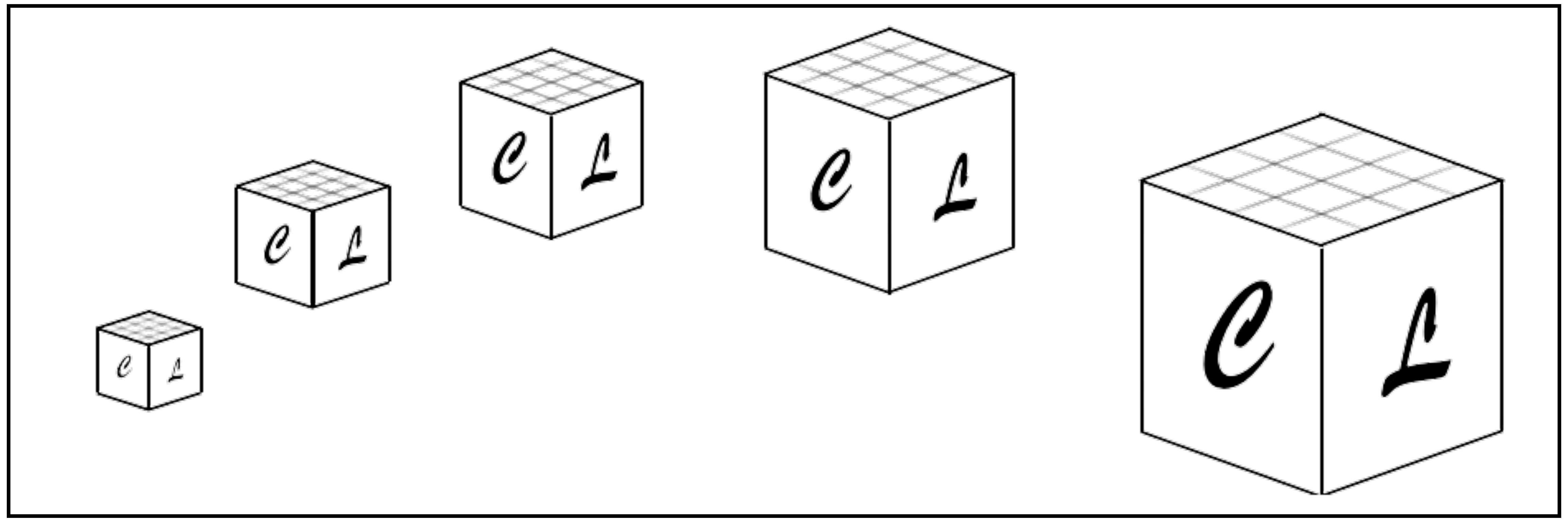


Simulating the Early Universe: CosmoLattice Workshop

Lecture 2: Introduction to Lattice Field Theory

Kenneth Marschall
(IFIC)

slides compiled with
Francisco Torrenti and Daniel Figueroa

Outline

I. Lattice

1. The Lattice
2. Boundary Conditions
3. Fourier Lattice
4. Lattice Derivatives
5. Power Spectrum

II. Scalar Field Dynamics on the Lattice

6. Discretising the Equations of Motion
7. Evolving the Scalar Field Equation of Motion
8. Initial Conditions

I. Lattice

1. The Lattice

1. The Lattice

Lattice:
Set of regularly spaced discrete coordinates

1D Lattice

1. The Lattice

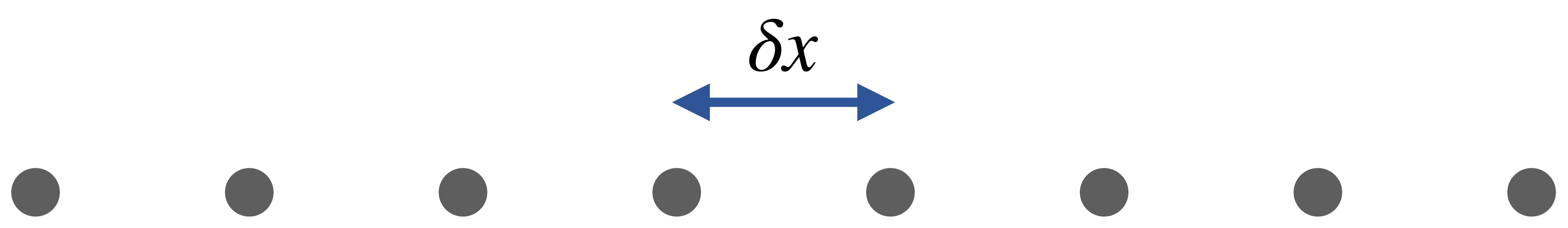
Lattice:
Set of regularly spaced discrete coordinates

Lattice is characterised by:

- dimensions: d
- number of lattice points: N^d
- lattice spacing: δx

1D Lattice

$N = 8$



1. The Lattice

Lattice:
Set of regularly spaced discrete coordinates

Lattice is characterised by:

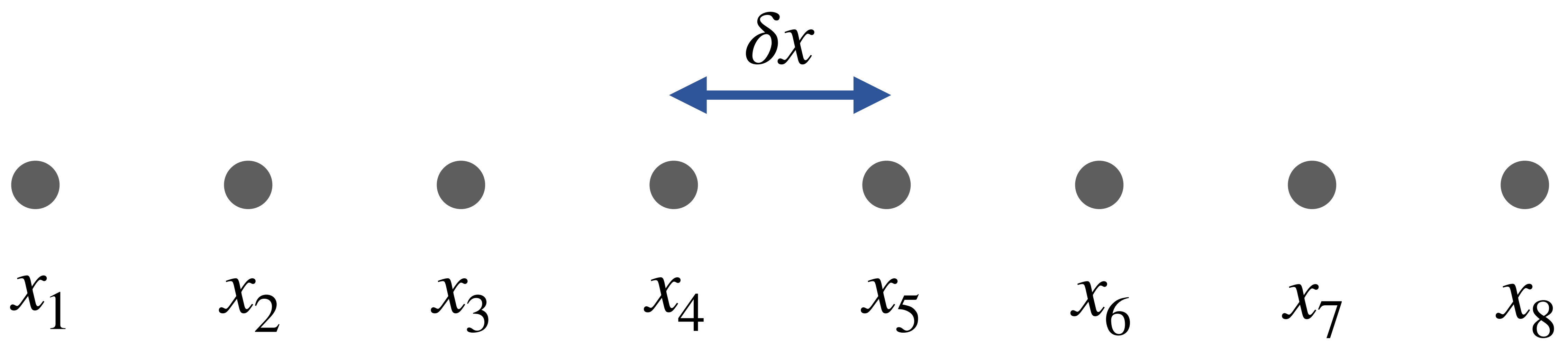
- dimensions: d
- number of lattice points: N^d
- lattice spacing: δx

Coordinates are tagged by:

$$\{x_n\}\,, \quad n = 1, \ldots, N$$

1D Lattice

$N = 8$

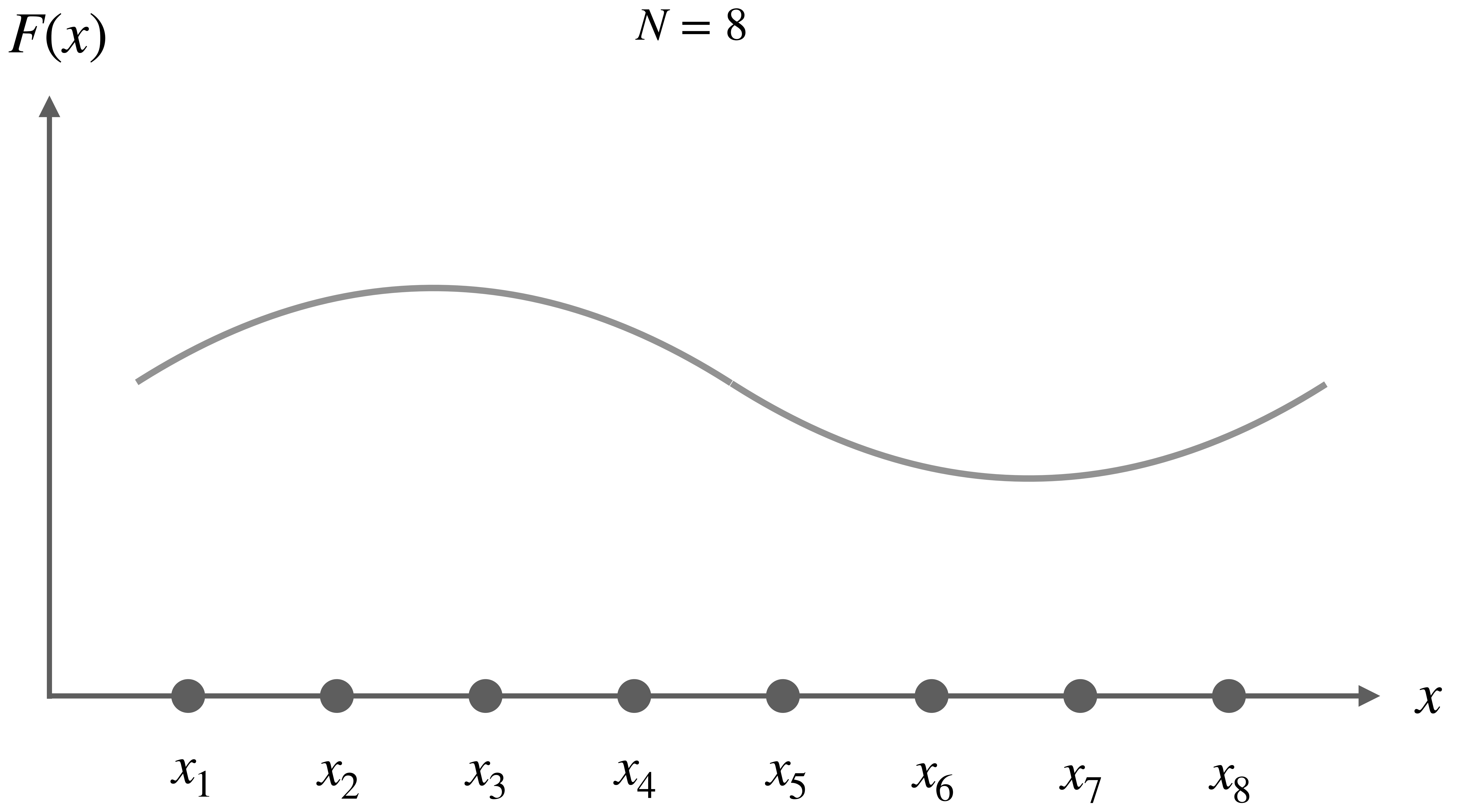


1. The Lattice

How do we represent a continuum function $F(x)$ on the lattice?

1D Lattice

$N = 8$



1. The Lattice

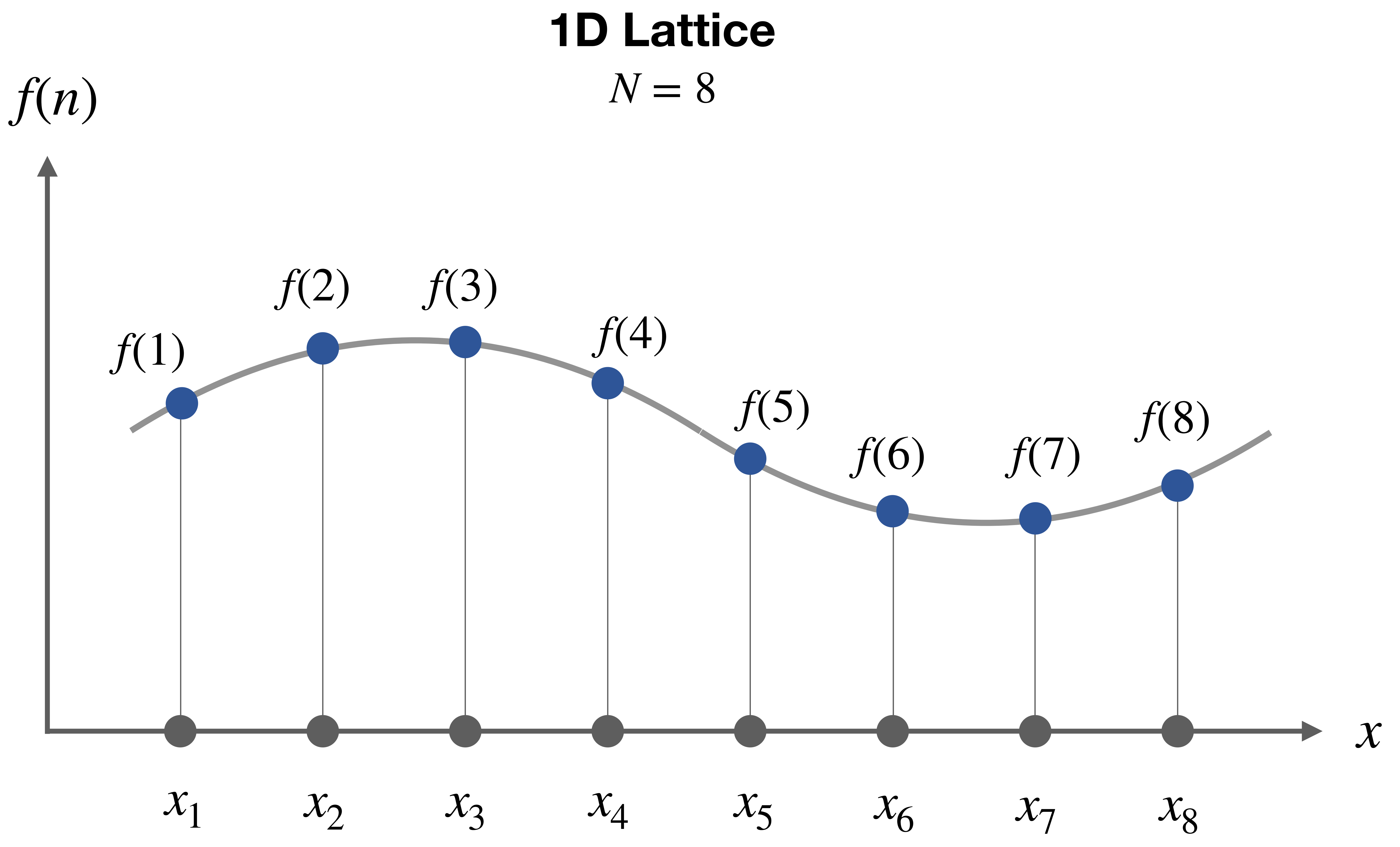
How do we represent a continuum function $F(x)$ on the lattice?

- define lattice function:

$$f(n) \equiv F(x_n \equiv x_* + n\delta x)$$

where $\quad n = 1, \ldots, N$

➔ the lattice function has the same as value continuum function at x_n



1. The Lattice

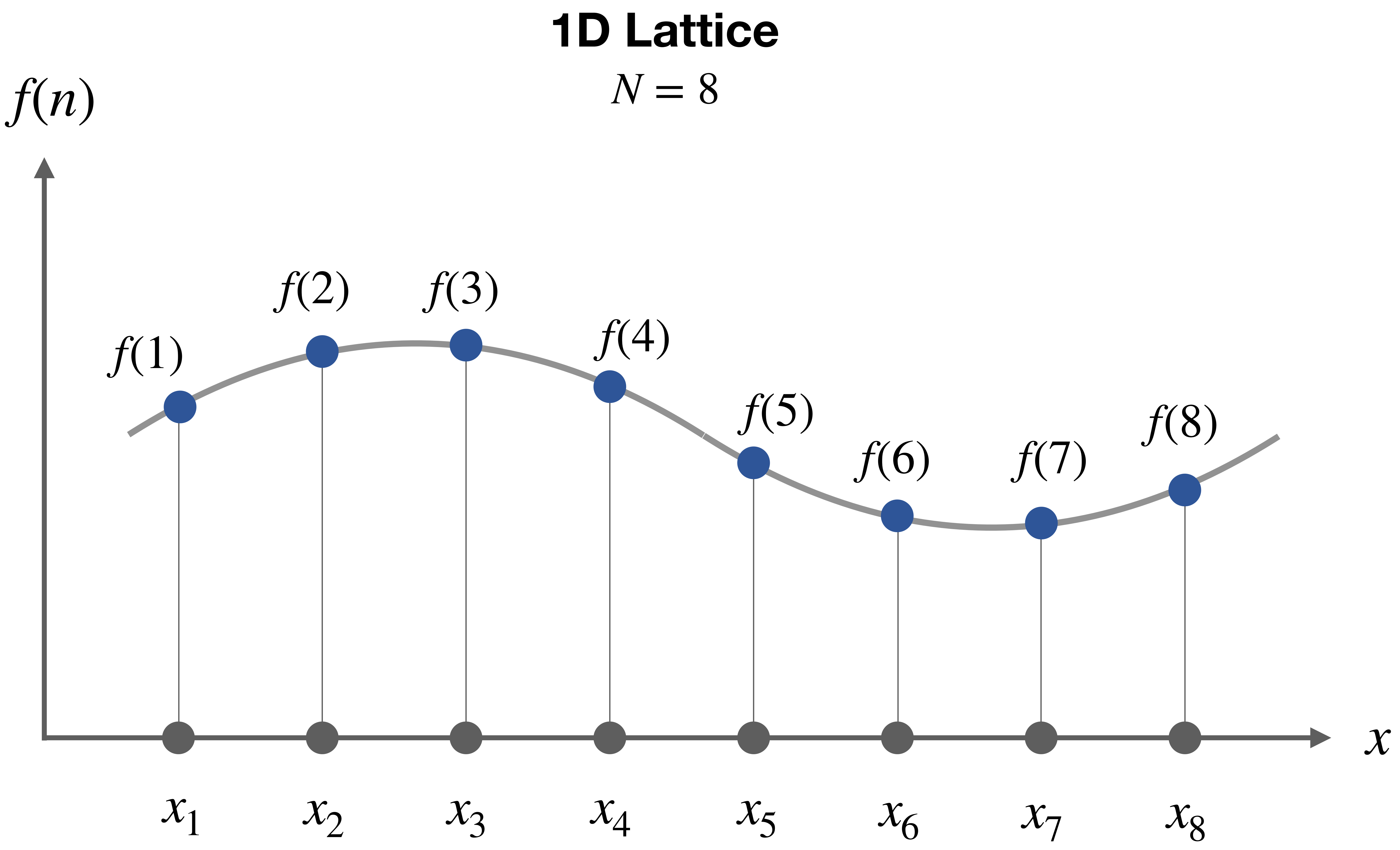
How do we represent a continuum function $F(x)$ on the lattice?

- define lattice function:

$$f(n) \equiv F(x_n \equiv x_* + n\delta x)$$

where $\quad n = 1, \ldots, N$

→ the lattice function has the same as value continuum function at x_n



How do we represent a continuum function $F(x)$ on the lattice?

- define lattice function:

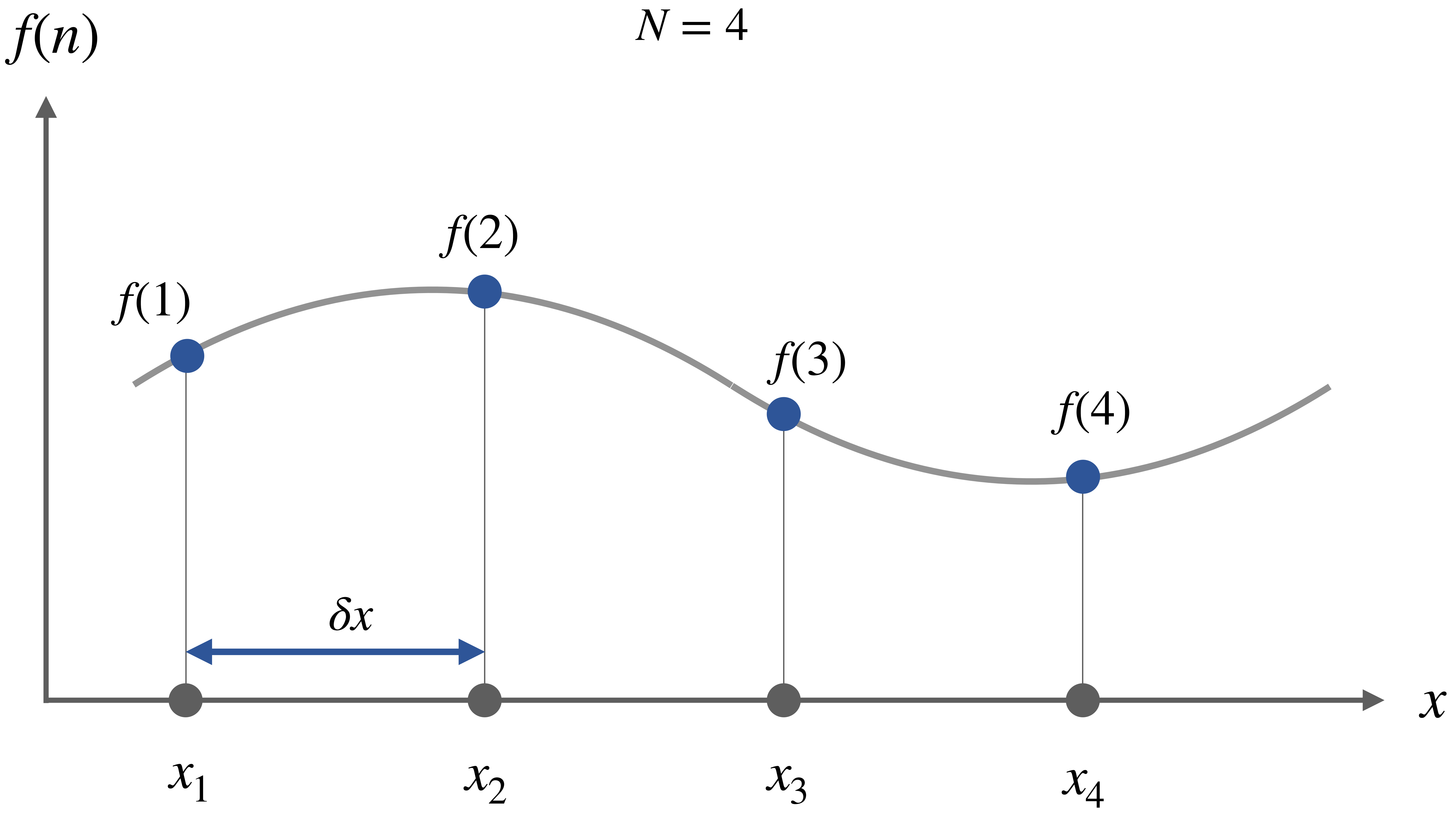
$$f(n) \equiv F(x_n \equiv x_* + n\delta x)$$

where $\quad n = 1, \ldots, N$

when is a representation of a function on the lattice good?

1D Lattice

$N = 4$



1. The Lattice

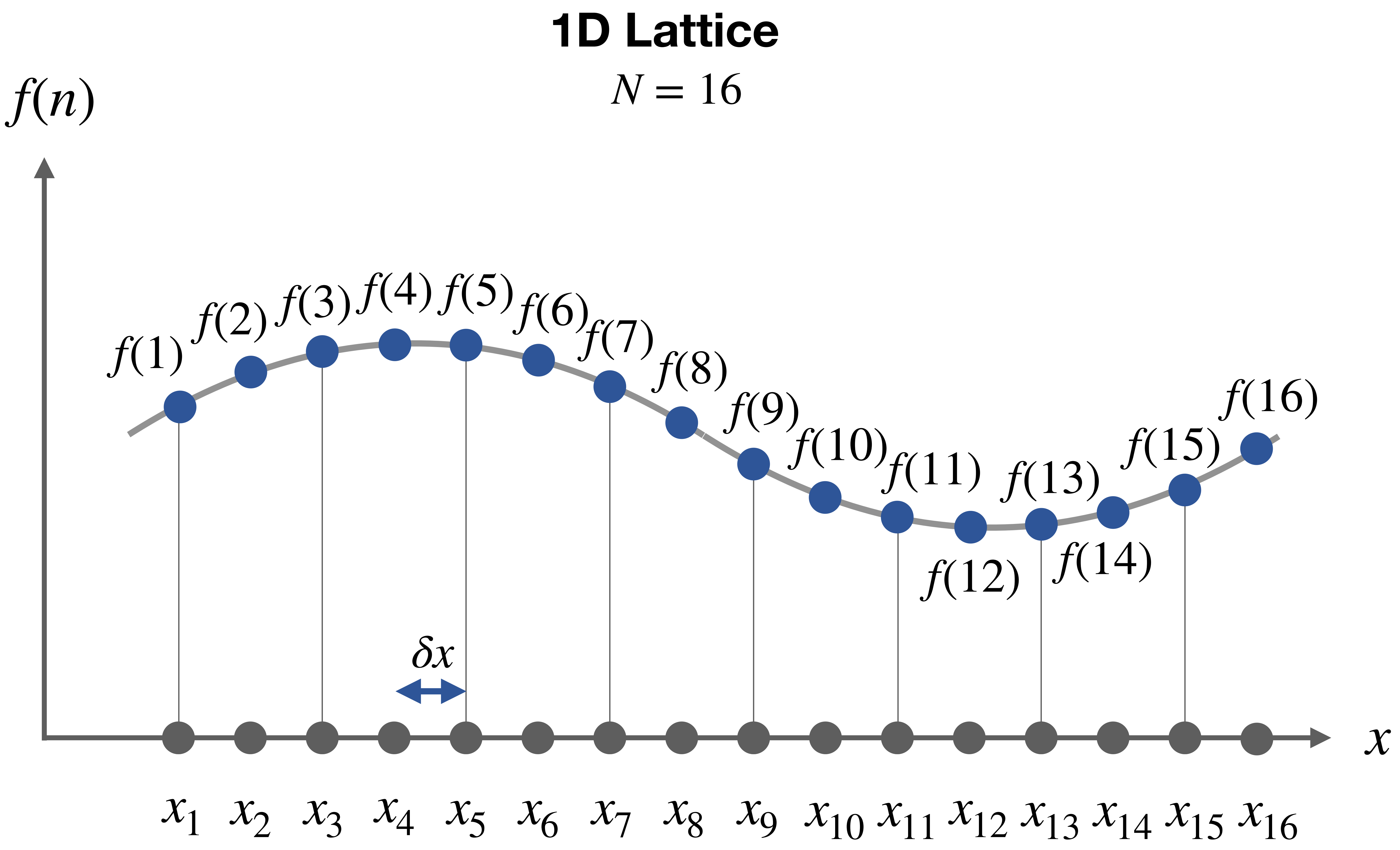
How do we represent a continuum function $F(x)$ on the lattice?

- define lattice function:

$$f(n) \equiv F(x_n \equiv x_* + n\delta x)$$

where $n = 1, \ldots, N$

when is a representation of a function on the lattice good?



1. The Lattice

How do we represent a continuum function $F(x)$ on the lattice?

- define lattice function:

$$f(n) \equiv F(x_n \equiv x_* + n\delta x)$$

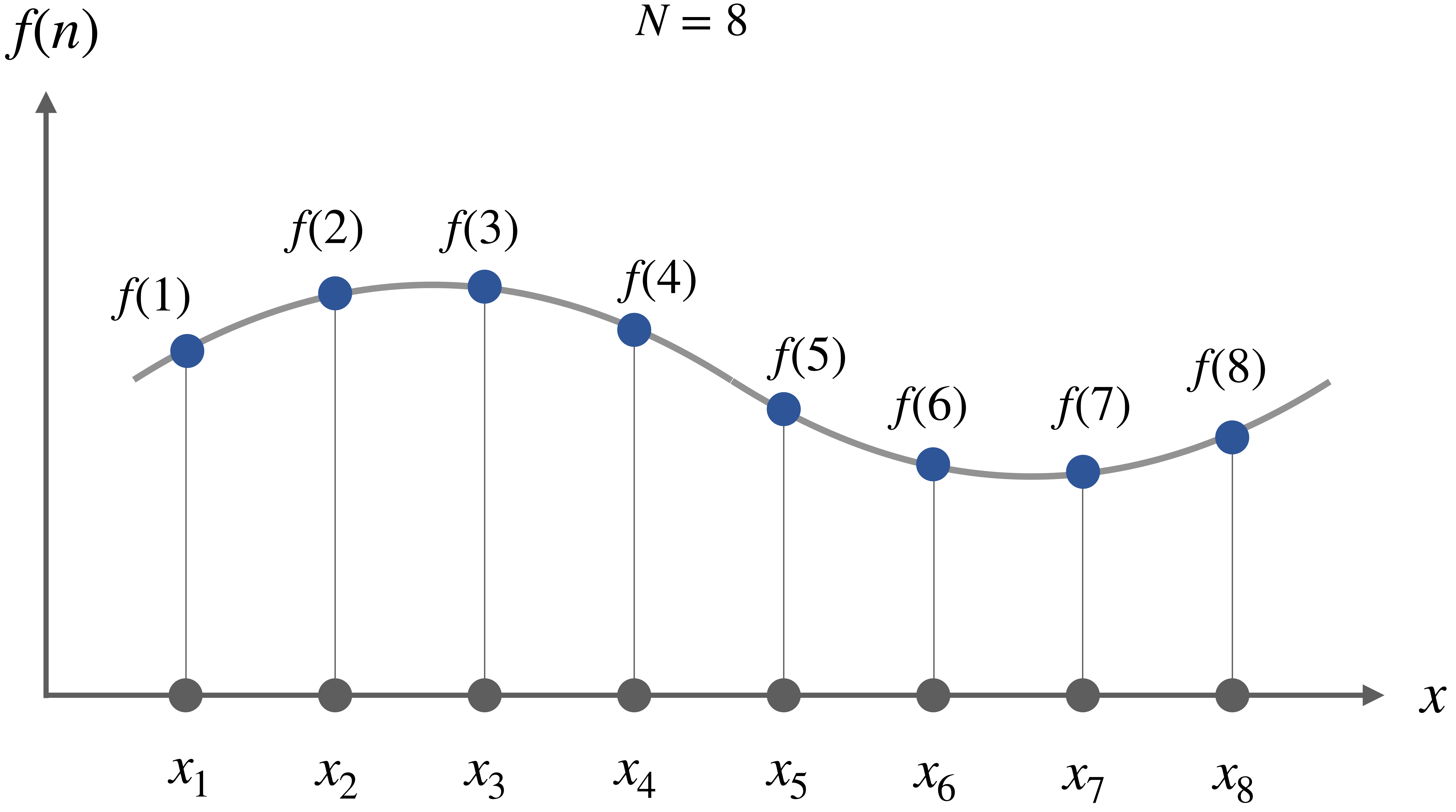
where $\quad n = 1, \ldots, N$

when is a representation of a function on the lattice good?

➔ $|f(n+1) - f(n)| \sim |F'(x_{n+\frac{1}{2}})| dx$

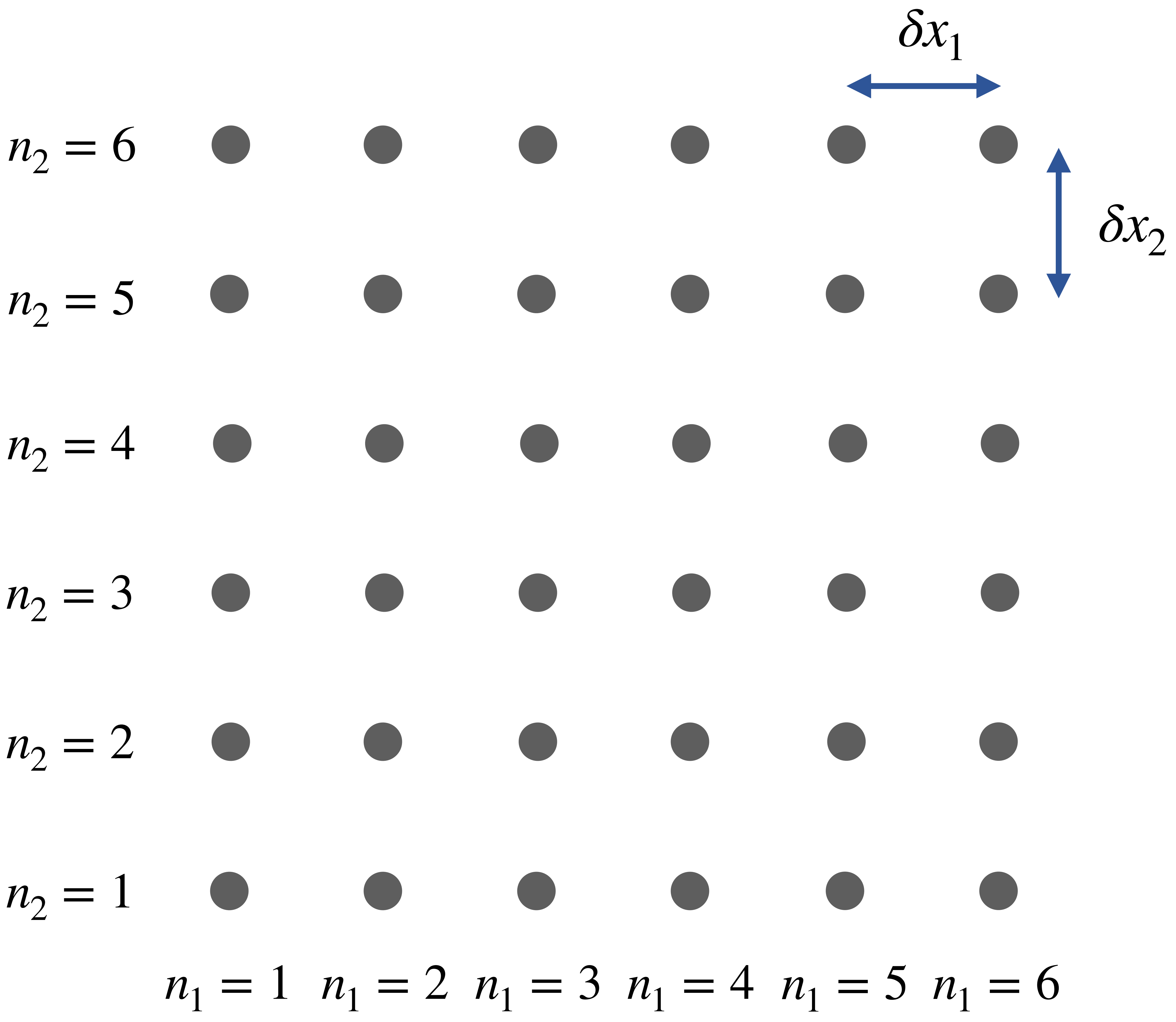
1D Lattice

$N = 8$



1. The Lattice

2D Lattice



1. The Lattice

Lattice is characterised by:

- number of lattice points: N^2
- lattice spacing: $\delta x_1 = \delta x_2$

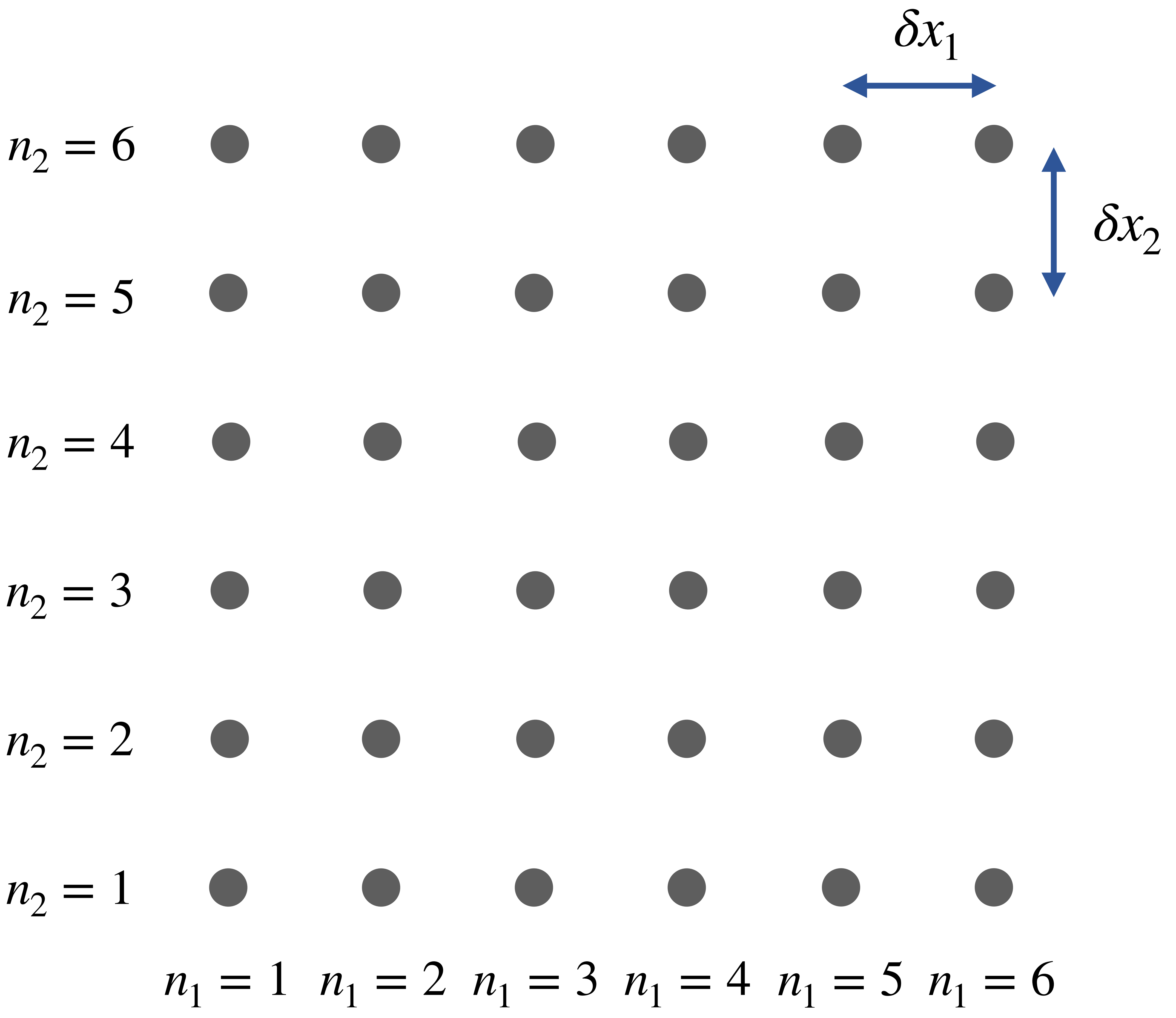
Coordinates:

$$\{\mathbf{x}_{n_1,n_2}\}\,, \quad n_i = 1,\, \ldots, N\,, \quad (i = 1, 2)$$

Lattice function:

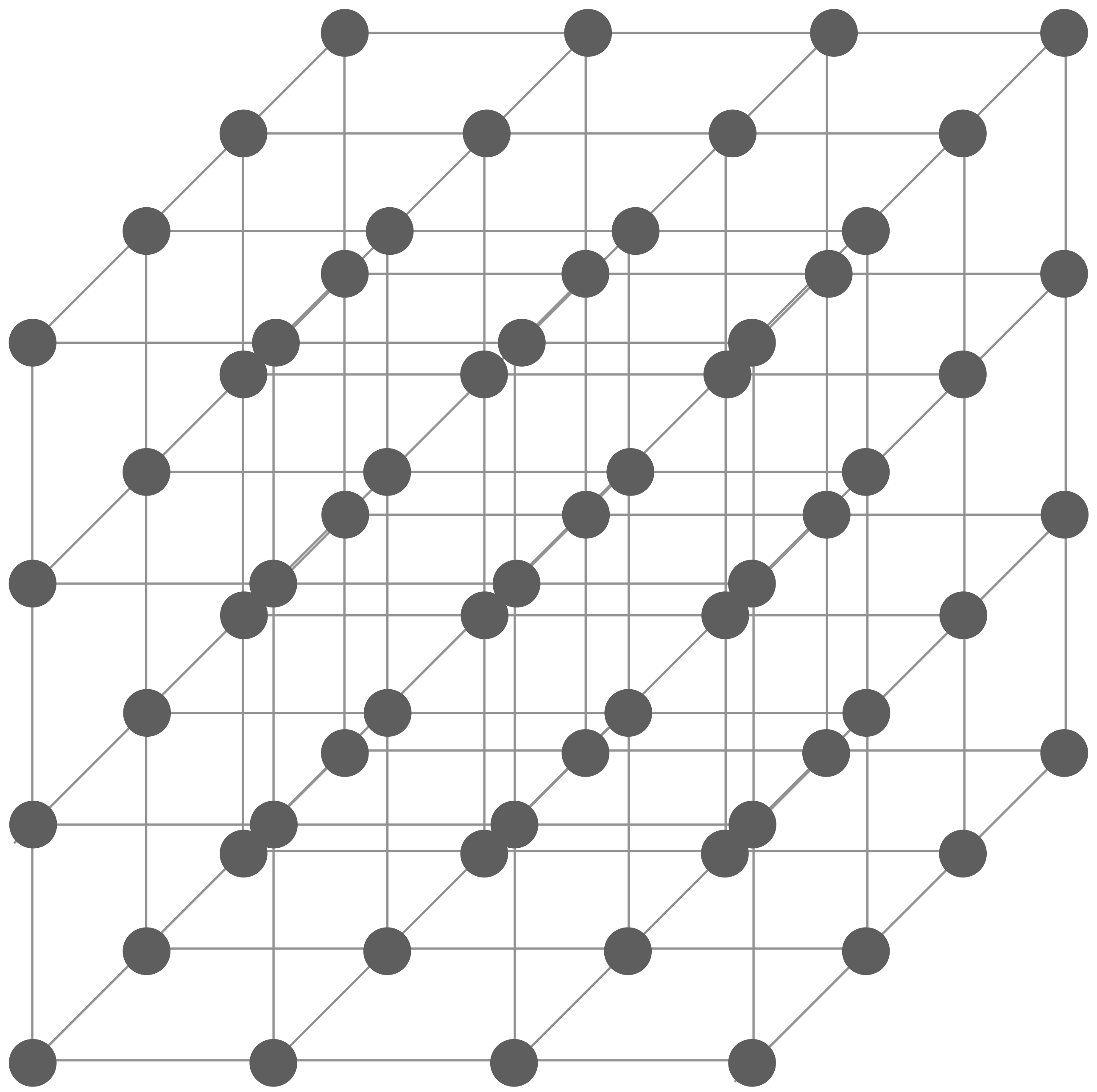
$$F(\mathbf{x}) \quad \longrightarrow \quad f(\mathbf{n}) = f(n_1, n_2) \equiv F(\mathbf{x}_{n_1 n_2})$$

2D Lattice



1. The Lattice

3D Lattice



Lattice is characterised by:

- number of lattice points: N^3
- lattice spacing: $\delta x_1 = \delta x_2 = \delta x_3$

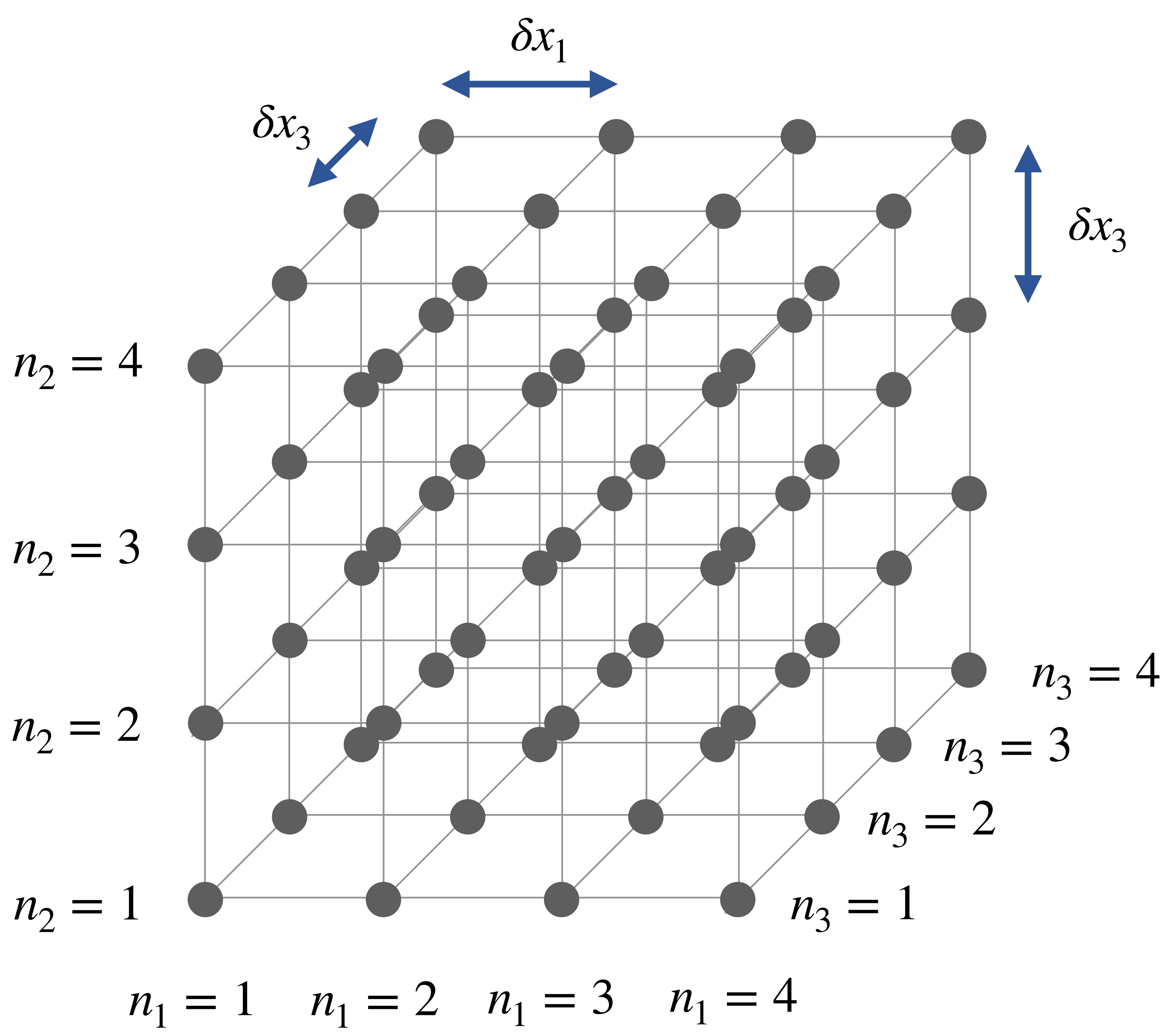
Coordinates:

$$\{\mathbf{x}_{n_1,n_2,n_3}\}\,,\quad n_i = 1,\,\ldots,N\,,\quad (i = 1,2,3)$$

Lattice function:

$$F(\mathbf{x}) \quad \longrightarrow \quad f(\mathbf{n}) = f(n_1, n_2, n_3) \equiv F(\mathbf{x}_{n_1 n_2 n_3})$$

3D Lattice



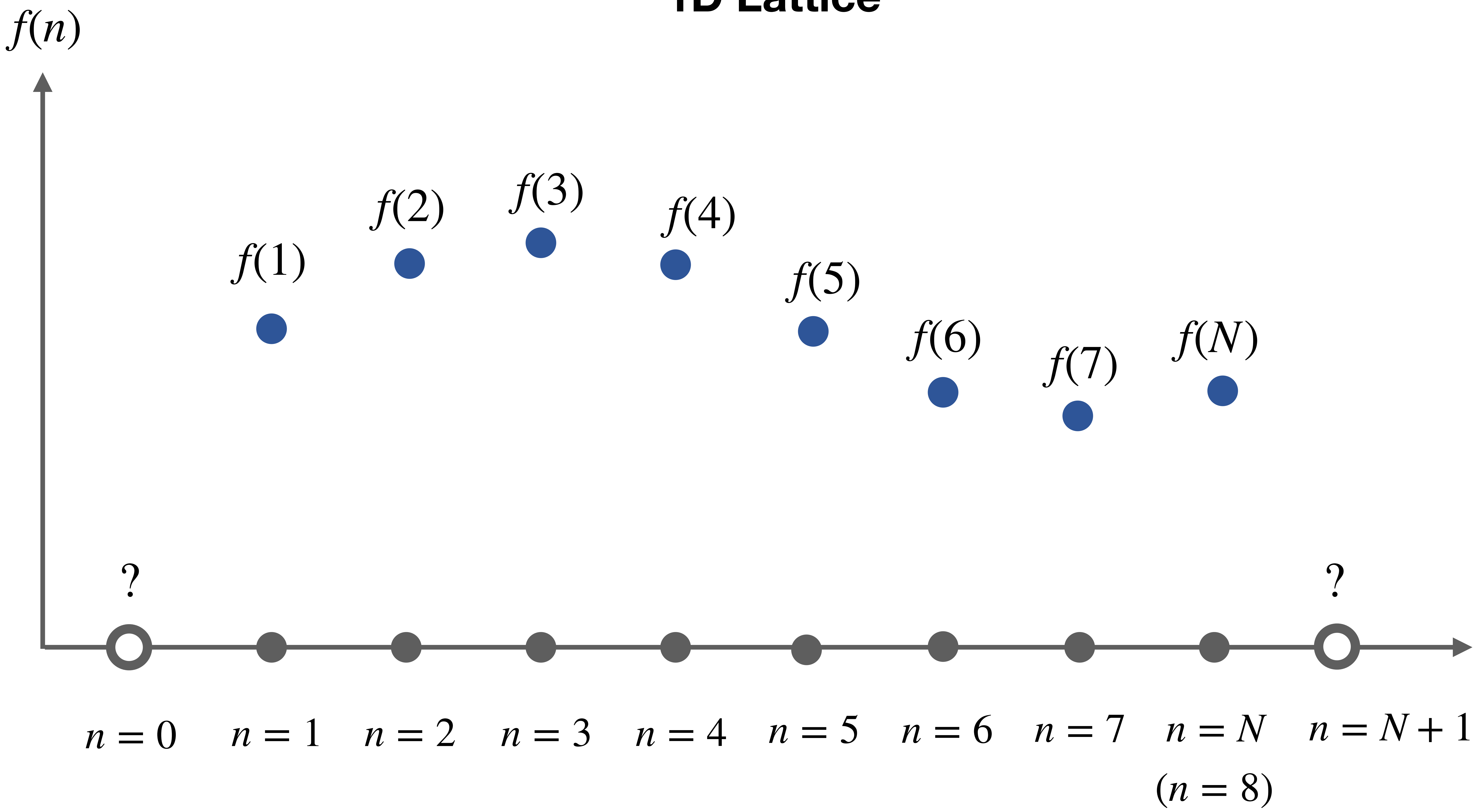
2. Boundary Conditions

2. Boundary Conditions

What about the boundaries?

What about the boundaries?

1D Lattice



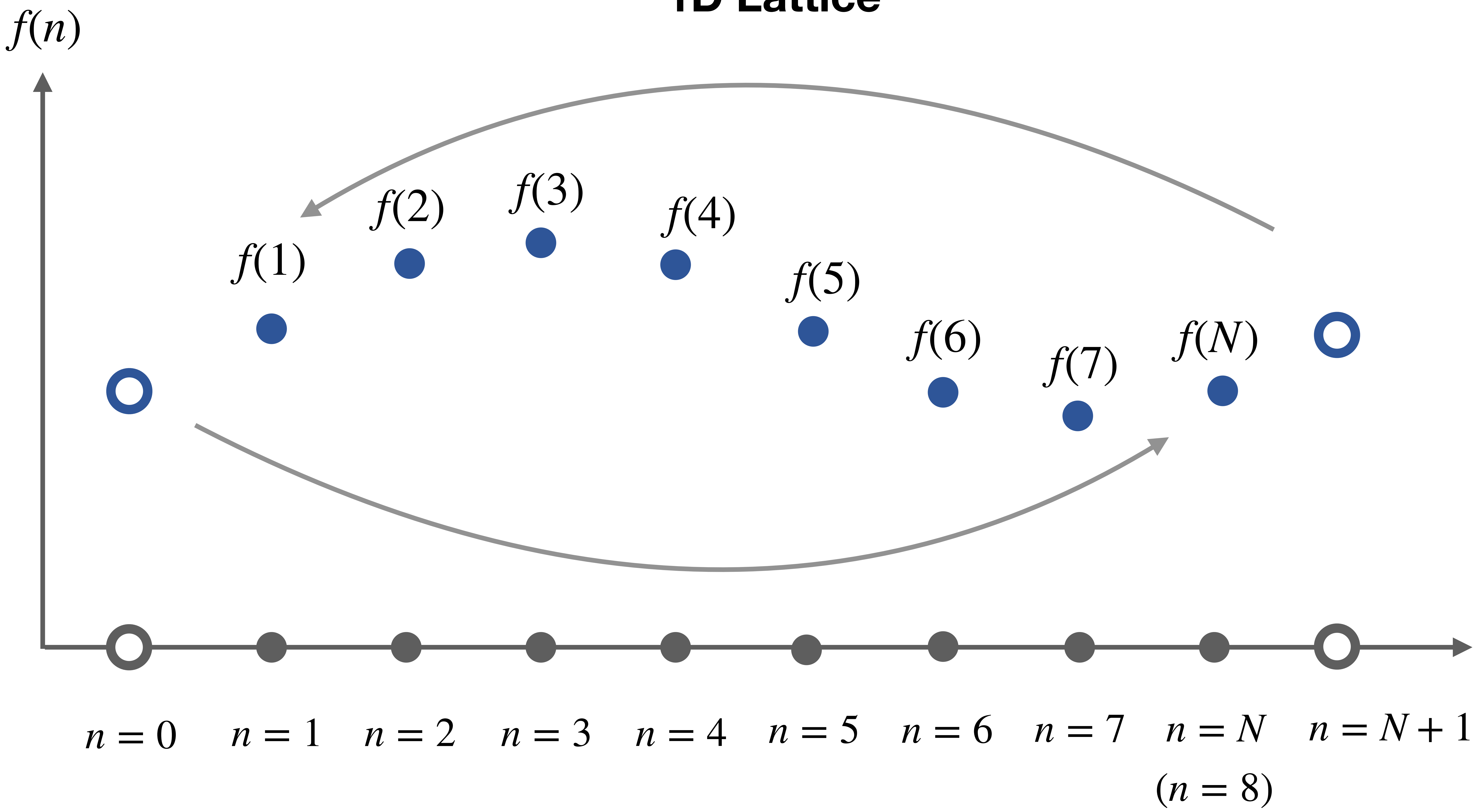
2. Boundary Conditions

What about the boundaries?

Periodic boundary conditions

- suitable for early universe phenomena

1D Lattice



2. Boundary Conditions

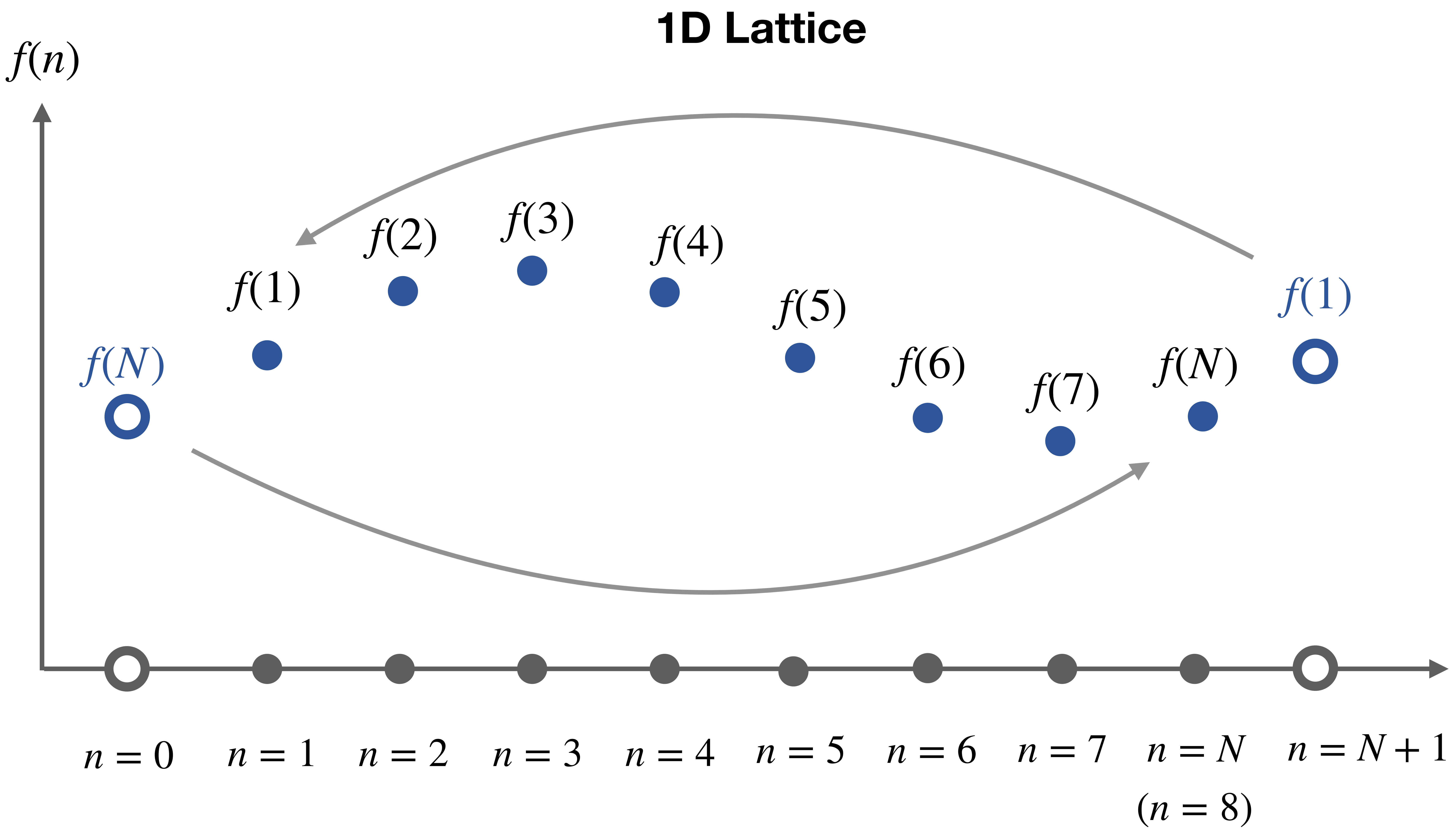
What about the boundaries?

Periodic boundary conditions

- suitable for early universe phenomena

$$\left\{ \begin{aligned} f(0,t) &\equiv f(N,t) \\ f(N+1,t) &\equiv f(1,t) \end{aligned} \right\}$$

for any $f(n,t) \equiv F(x_n,t)$
with $n = 1, \ldots, N$



Periodic boundary conditions in 3D

$$\mathbf{n} \equiv (n_1, n_2, n_3)\,, \qquad \{f(\mathbf{n})\}$$

$$n_i = 1,\ \ldots, N\,, \quad (i = 1, 2, 3)$$

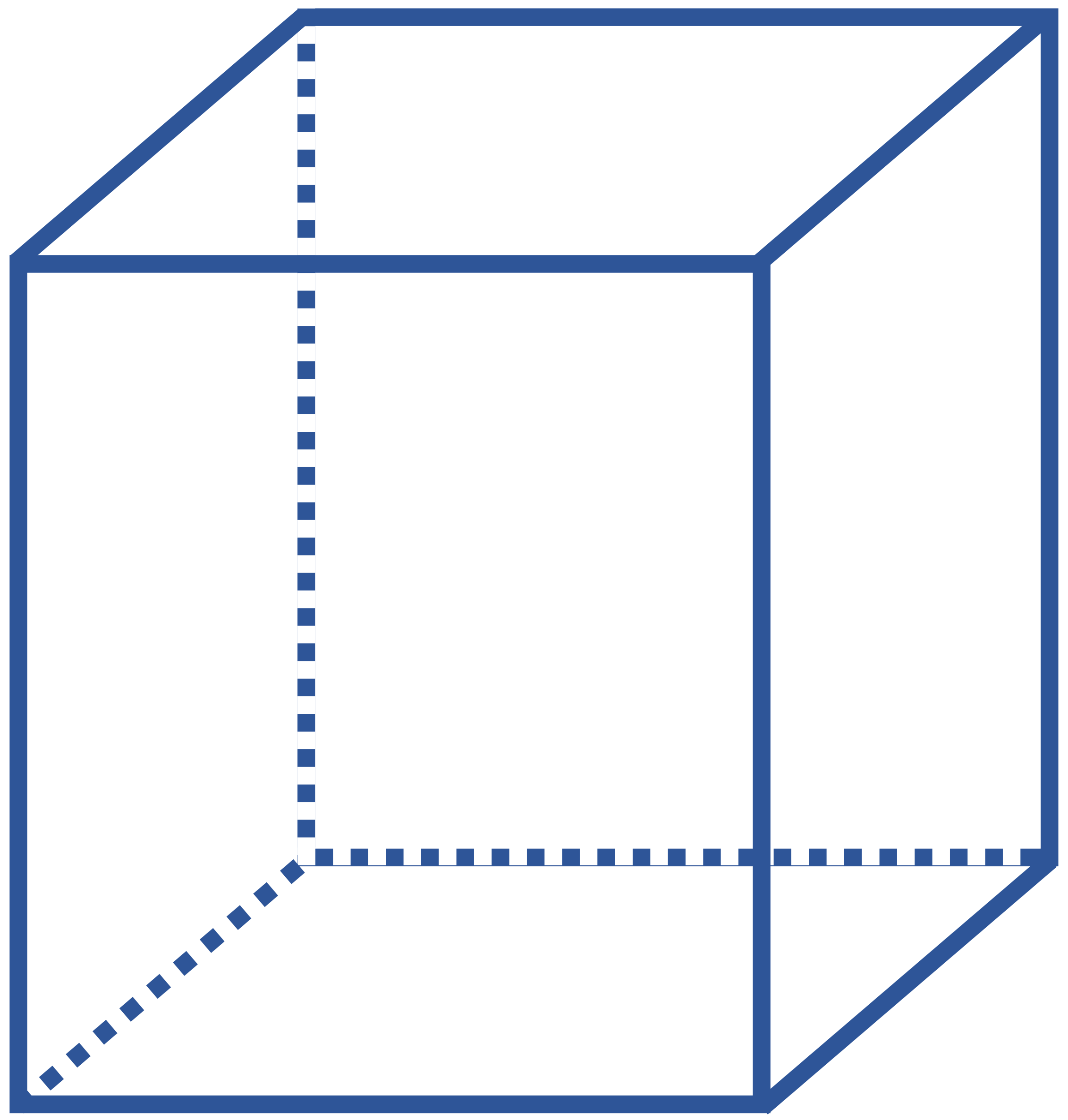
$\updownarrow$ periodic lattice

$$\boxed{f(\mathbf{n} + \hat{\imath}N) \equiv f(\mathbf{n})}$$

$\hat{\imath}$ ≡ unit vector in direction i, s.t.:

$$\hat{1} \equiv (1{,}0{,}0),\ \hat{2} \equiv (0{,}1{,}0)$$
$$\hat{3} \equiv (0{,}0{,}1)$$

3D Lattice



2. Boundary Conditions

Periodic boundary conditions in 3D

$$\mathbf{n} \equiv (n_1, n_2, n_3)\,, \qquad \{f(\mathbf{n})\}$$

$$n_i = 1,\ \ldots, N\,, \quad (i = 1, 2, 3)$$

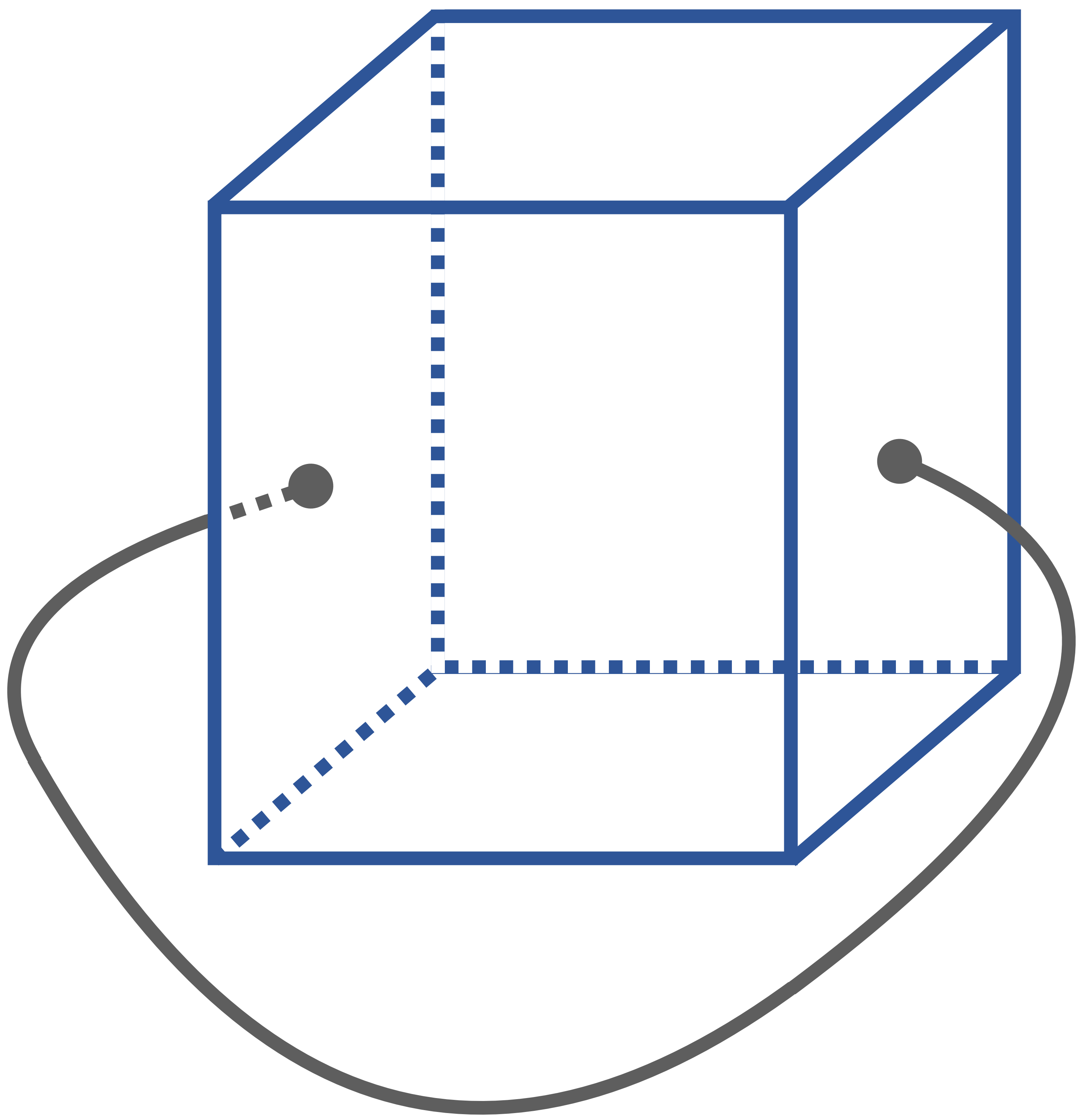
periodic
lattice

$$\boxed{f(\mathbf{n} + \hat{\imath}N) \equiv f(\mathbf{n})}$$

$\hat{\imath} \equiv$ unit vector in direction i, s.t.:

$$\hat{1} \equiv (1{,}0{,}0),\ \hat{2} \equiv (0{,}1{,}0)$$
$$\hat{3} \equiv (0{,}0{,}1)$$

3D Lattice



Periodic boundary conditions in 3D

$$\mathbf{n} \equiv (n_1, n_2, n_3)\,, \qquad \{f(\mathbf{n})\}$$

$$n_i = 1,\ \ldots, N\,, \quad (i = 1, 2, 3)$$

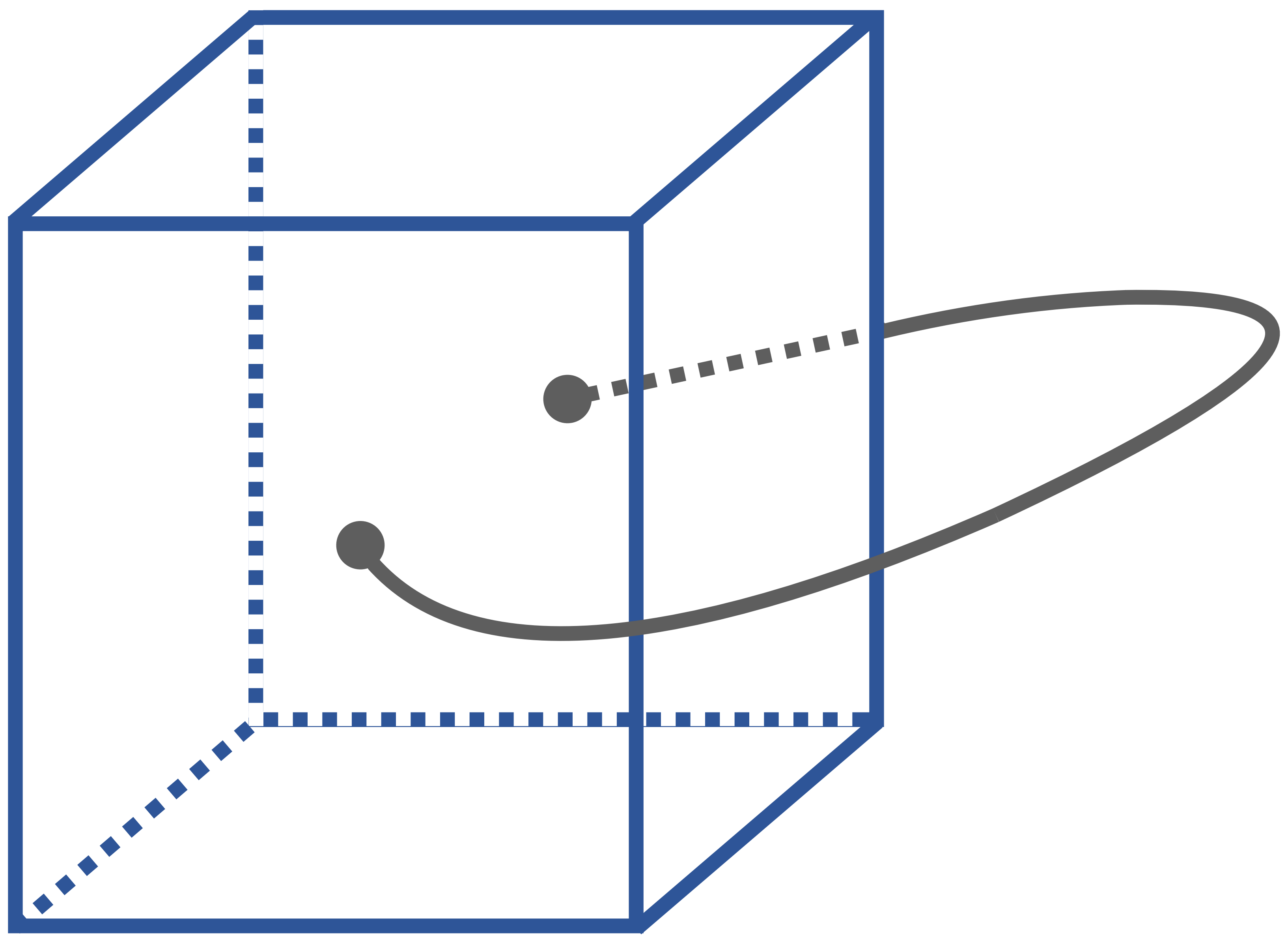
$\Updownarrow$ periodic lattice

$$\boxed{f(\mathbf{n} + \hat{\imath} N) \equiv f(\mathbf{n})}$$

$\hat{\imath} \equiv$ unit vector in direction i, s.t.:

$$\hat{1} \equiv (1{,}0{,}0),\ \hat{2} \equiv (0{,}1{,}0)$$
$$\hat{3} \equiv (0{,}0{,}1)$$

3D Lattice



Periodic boundary conditions in 3D

$$\mathbf{n} \equiv (n_1, n_2, n_3)\,, \qquad \{f(\mathbf{n})\}$$

$$n_i = 1,\ \ldots, N\,, \quad (i = 1, 2, 3)$$

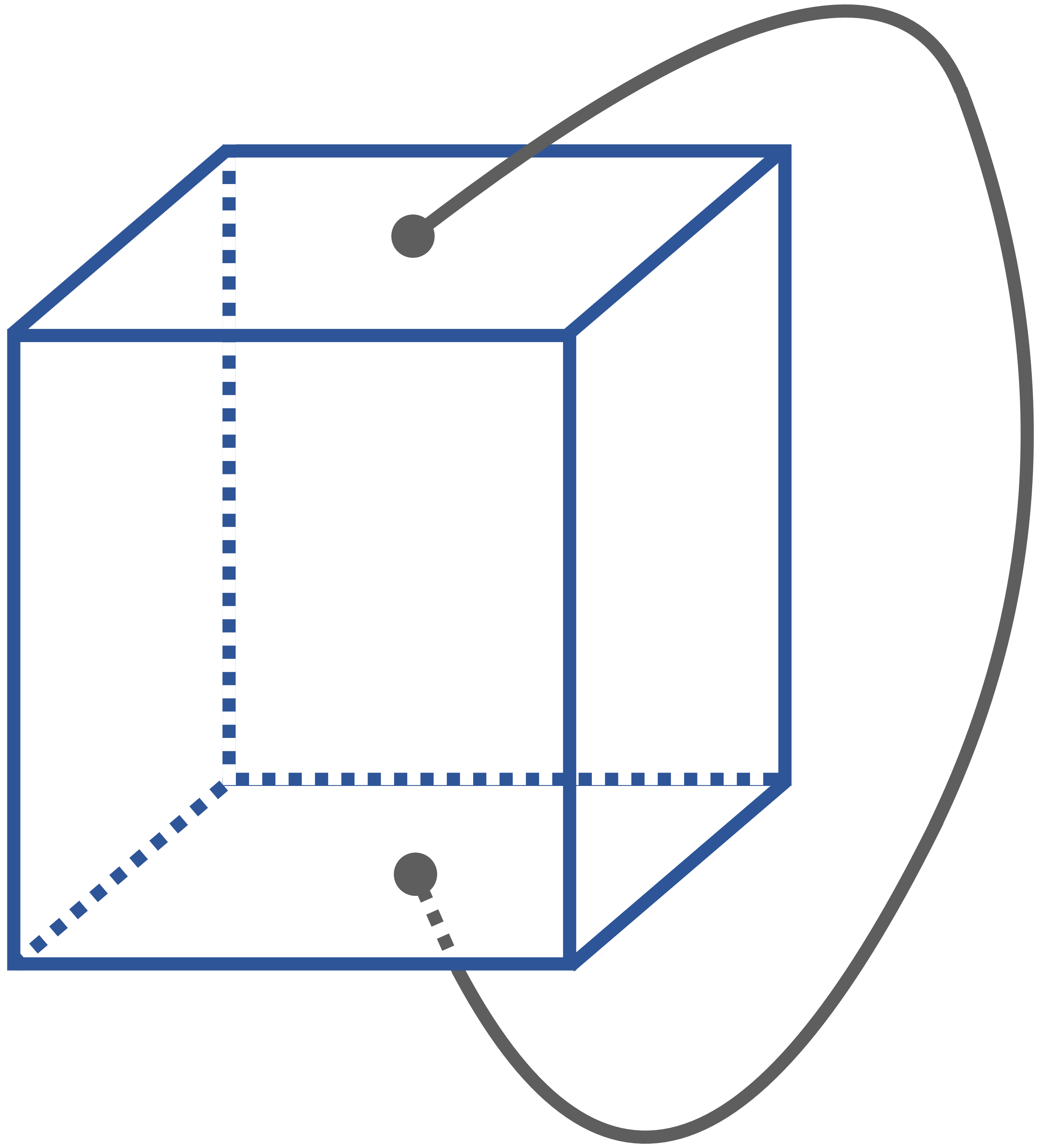
periodic lattice

$$\boxed{f(\mathbf{n} + \hat{\imath}N) \equiv f(\mathbf{n})}$$

$\hat{\imath} \equiv$ unit vector in direction i, s.t.:

$$\hat{1} \equiv (1{,}0{,}0),\ \hat{2} \equiv (0{,}1{,}0)$$
$$\hat{3} \equiv (0{,}0{,}1)$$

3D Lattice



2. Fourier Lattice

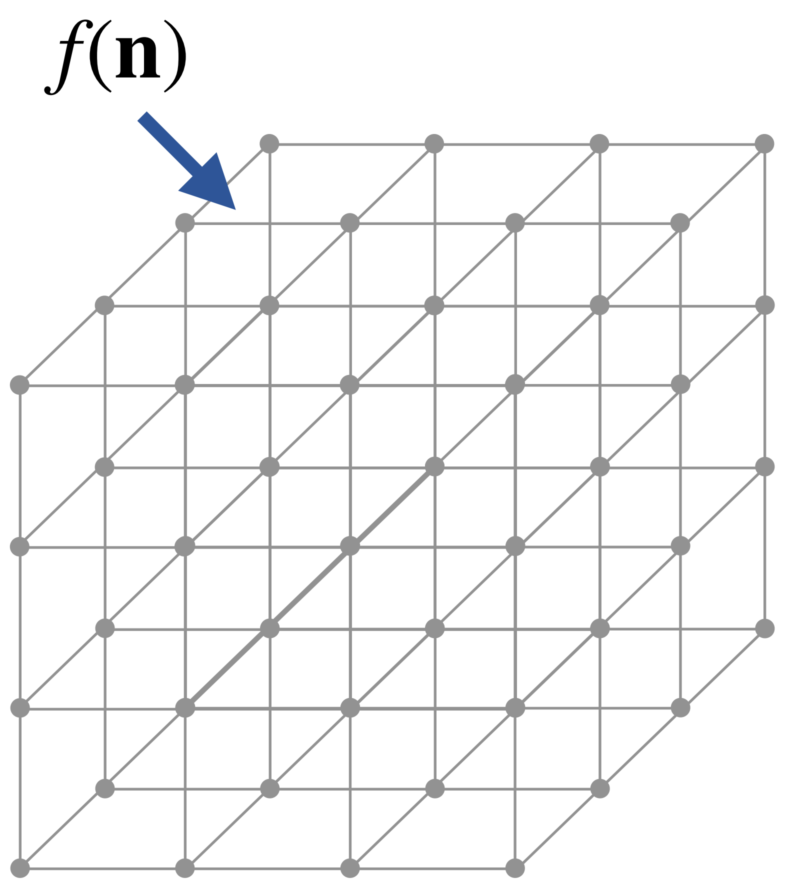
Discrete Fourier Transform:

$$f(\mathbf{n}) \equiv \frac{1}{N^3} \sum_{\tilde{\mathbf{n}}} e^{+i\frac{2\pi}{N}\tilde{\mathbf{n}}\mathbf{n}} f(\tilde{\mathbf{n}}) \quad \longleftrightarrow \quad f(\tilde{\mathbf{n}}) \equiv \sum_{\mathbf{n}} e^{-i\frac{2\pi}{N}\mathbf{n}\tilde{\mathbf{n}}} f(\mathbf{n})$$

Discrete Fourier Transform:

$$f(\mathbf{n}) \equiv \frac{1}{N^3} \sum_{\tilde{\mathbf{n}}} e^{+i\frac{2\pi}{N}\tilde{\mathbf{n}}\mathbf{n}} f(\tilde{\mathbf{n}}) \quad \longleftrightarrow \quad f(\tilde{\mathbf{n}}) \equiv \sum_{\mathbf{n}} e^{-i\frac{2\pi}{N}\mathbf{n}\tilde{\mathbf{n}}} f(\mathbf{n})$$

Real Lattice:



- discrete positions:

$$\mathbf{n} \equiv (n_1, n_2, n_3)\,,$$
$$n_i = 1, \ \ldots, N$$

- periodic boundaries:

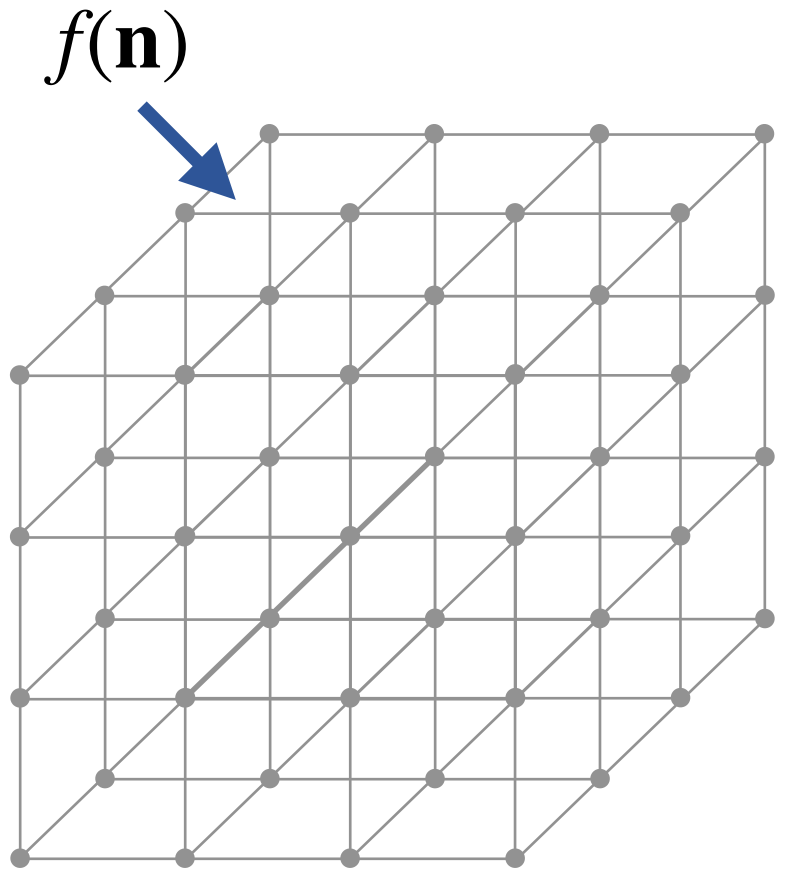
$$f(\mathbf{n} + \hat{\imath}N) \equiv f(\mathbf{n})$$

3. Fourier Lattice

Discrete Fourier Transform:

$$f(\mathbf{n}) \equiv \frac{1}{N^3} \sum_{\tilde{\mathbf{n}}} e^{+i\frac{2\pi}{N}\tilde{\mathbf{n}}\mathbf{n}} f(\tilde{\mathbf{n}}) \quad \longleftrightarrow \quad f(\tilde{\mathbf{n}}) \equiv \sum_{\mathbf{n}} e^{-i\frac{2\pi}{N}\mathbf{n}\tilde{\mathbf{n}}} f(\mathbf{n})$$

Real Lattice:



- discrete positions:

$$\mathbf{n} \equiv (n_1, n_2, n_3)\,,$$
$$n_i = 1, \;\ldots, N$$

- periodic boundaries:

$$f(\mathbf{n} + \hat{\imath}N) \equiv f(\mathbf{n})$$

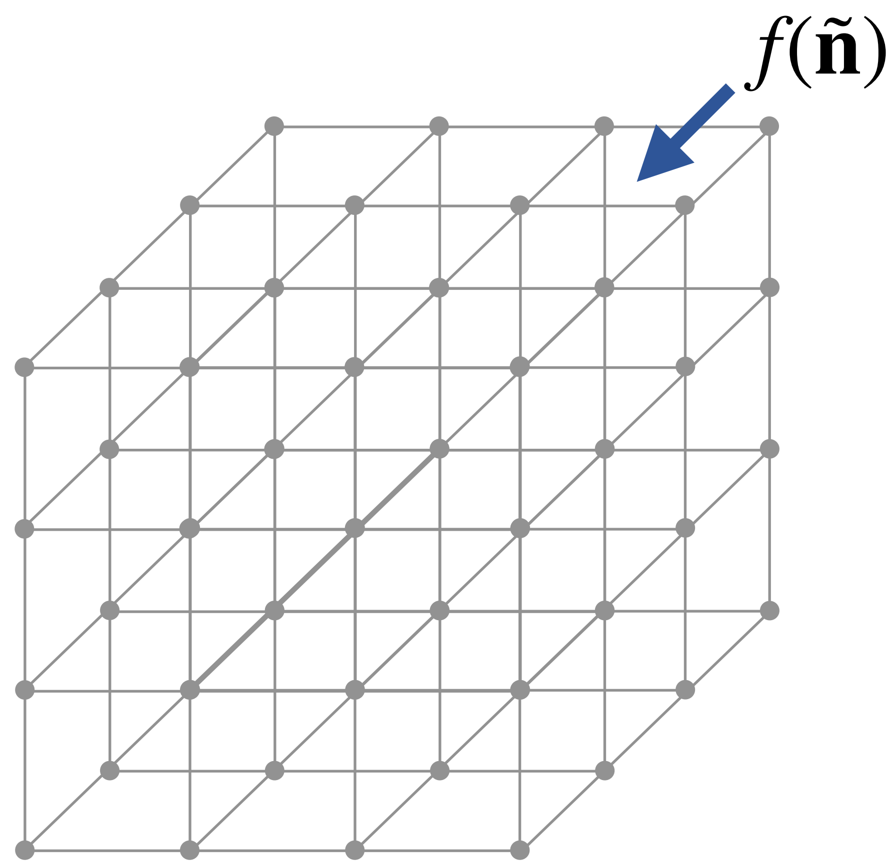
Fourier Lattice:

- discrete momenta:

$$\tilde{\mathbf{n}} \equiv (\tilde{n}_1, \tilde{n}_2, \tilde{n}_3)\,,$$
$$\tilde{n}_i = -\frac{N}{2} + 1, \;\ldots\;, -1, 0, 1, \;\ldots\;, \frac{N}{2}$$

- periodic Fourier modes:

$$f(\tilde{\mathbf{n}} + \hat{\imath}N) \equiv f(\tilde{\mathbf{n}})$$



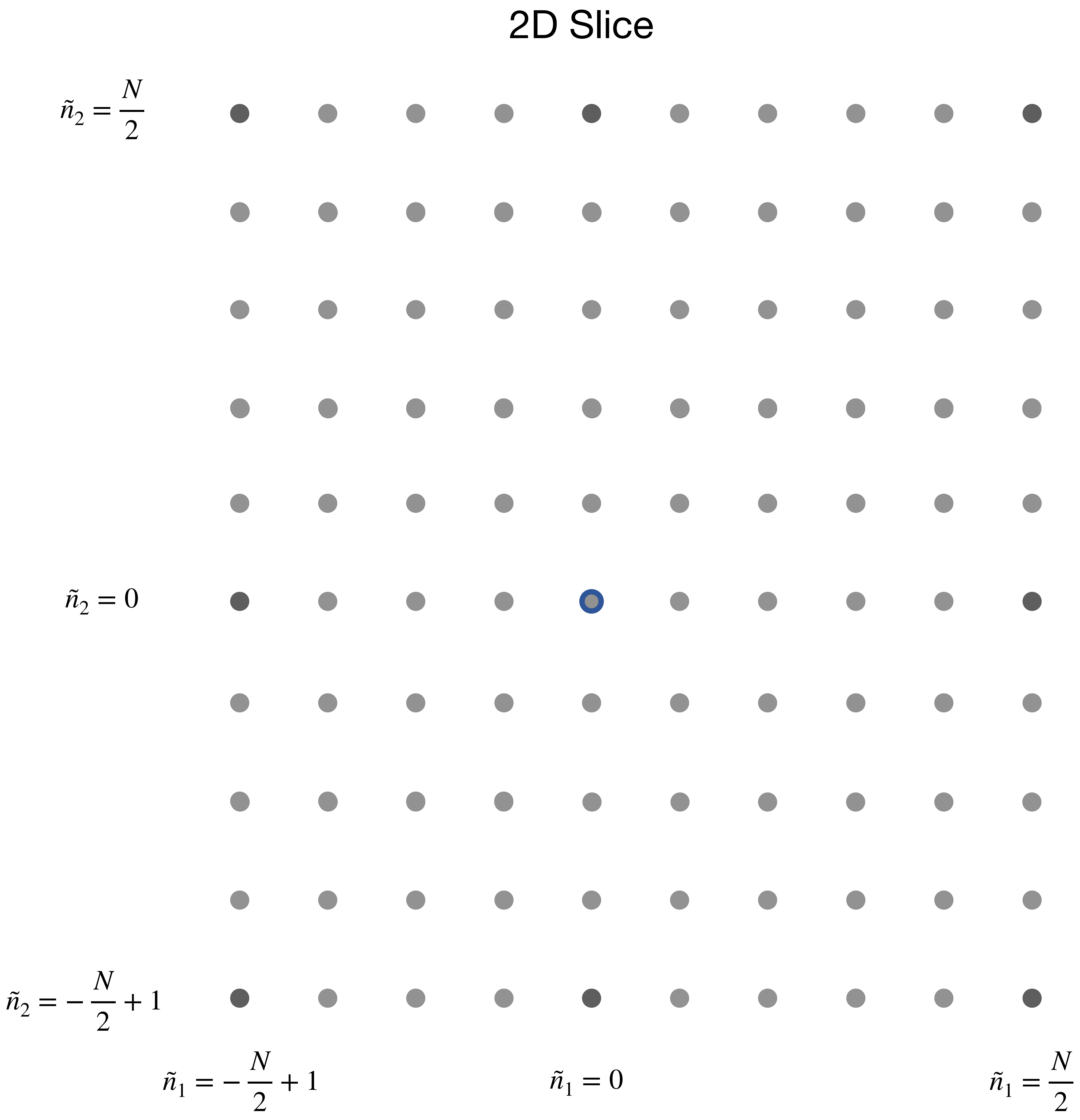
Fourier Lattice:

$$\tilde{n}_i = -\frac{N}{2} + 1, \; \dots \;, -1, 0, 1, \; \dots \;, \frac{N}{2}$$

FL captures **physical momenta:**

$$\mathbf{k}(\tilde{\mathbf{n}}) \equiv \frac{2\pi}{L}\tilde{\mathbf{n}}$$

$$\rightarrow \quad (k_1, k_2, k_3) = \frac{2\pi}{L}(\tilde{n}_1, \tilde{n}_2, \tilde{n}_3)$$



2. Fourier Lattice

Fourier Lattice:

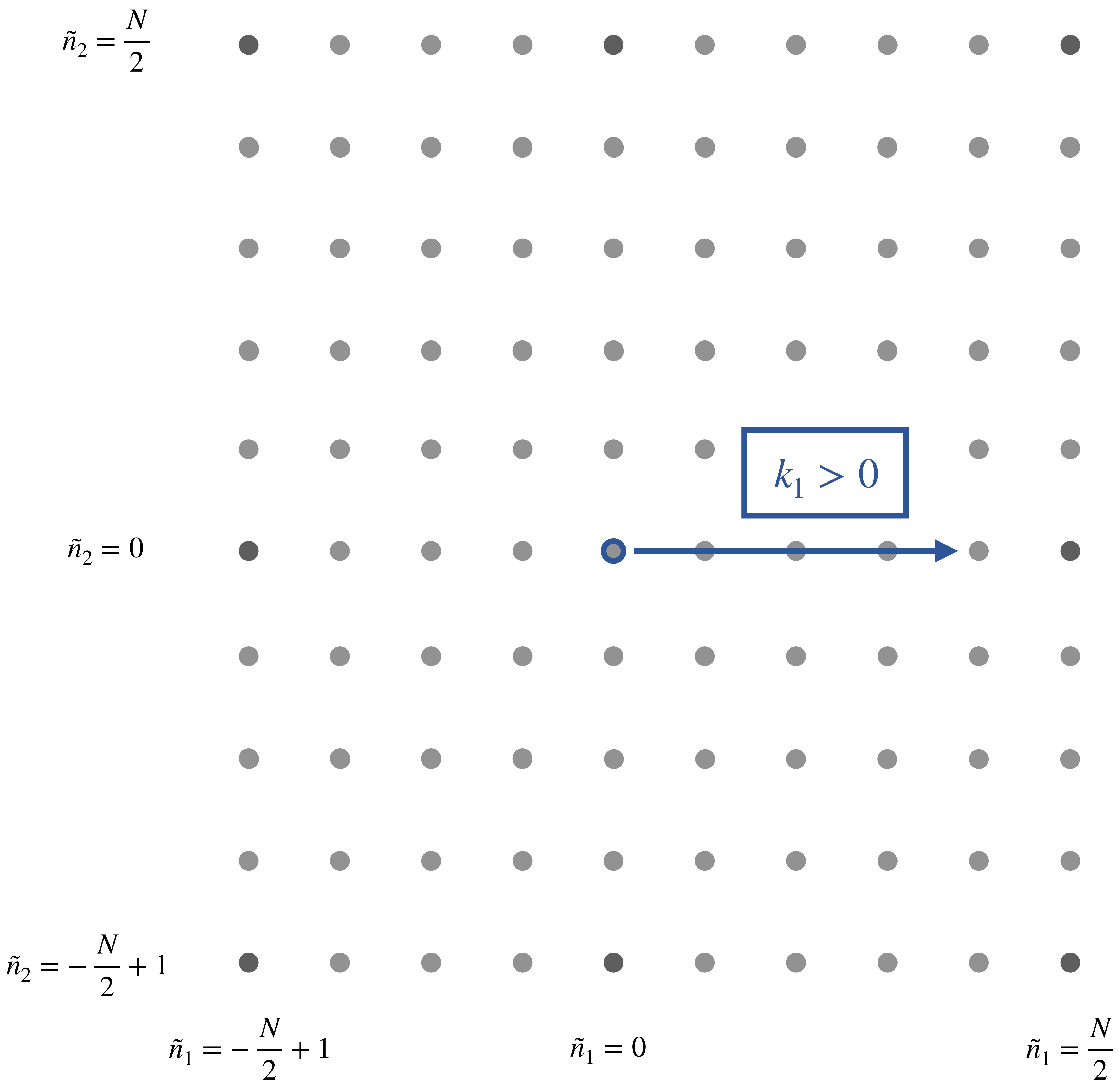
$$\tilde{n}_i = -\frac{N}{2}+1, \ \dots \ , -1, 0, 1, \ \dots \ , \frac{N}{2}$$

FL captures **physical momenta:**

$$\mathbf{k}(\tilde{\mathbf{n}}) \equiv \frac{2\pi}{L}\tilde{\mathbf{n}}$$

$$\rightarrow \quad (k_1, k_2, k_3) = \frac{2\pi}{L}(\tilde{n}_1, \tilde{n}_2, \tilde{n}_3)$$

2D Slice



Fourier Lattice:

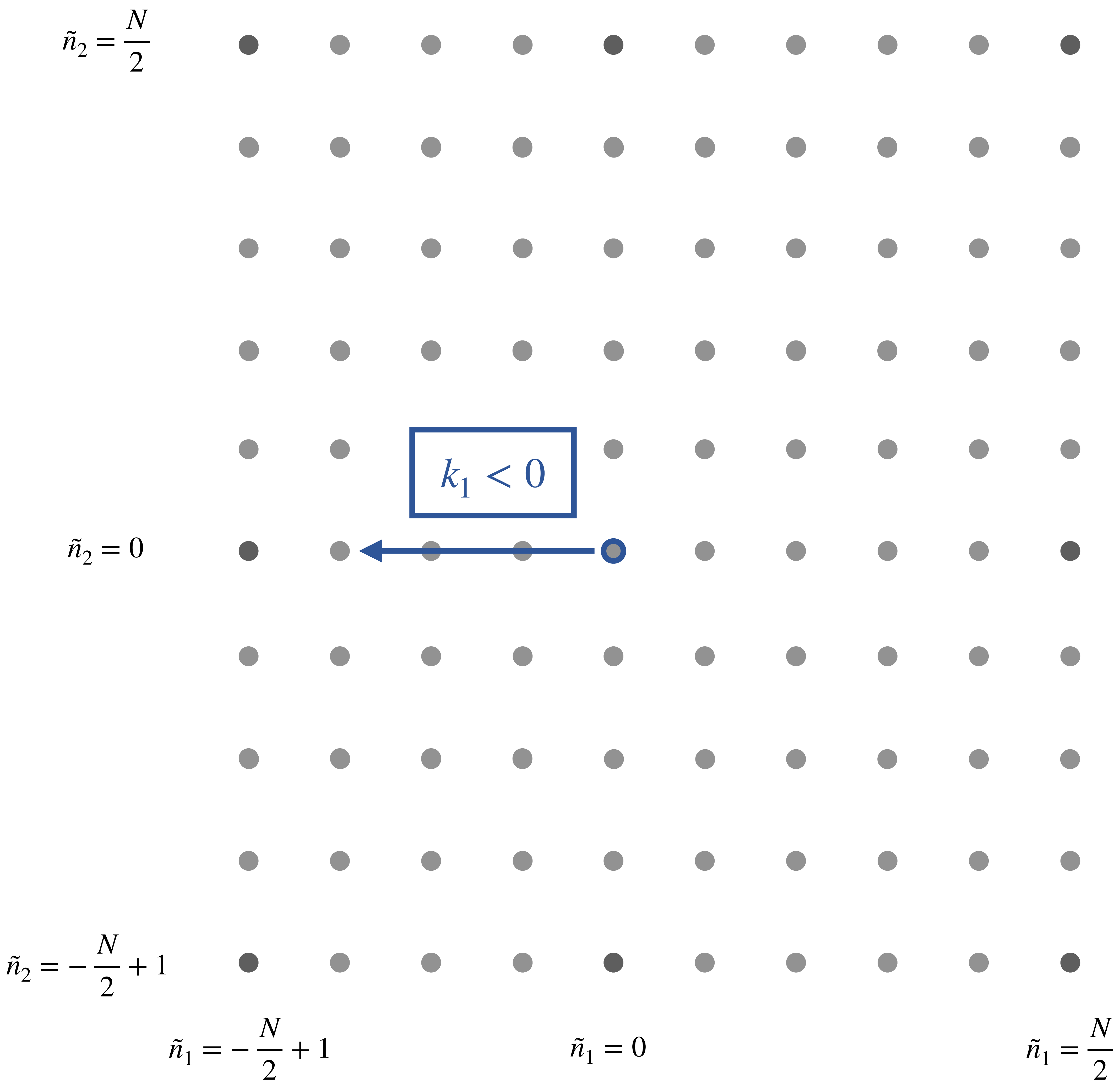
$$\tilde{n}_i = -\frac{N}{2} + 1, \ \dots \ , -1, 0, 1, \ \dots \ , \frac{N}{2}$$

FL captures **physical momenta:**

$$\mathbf{k}(\tilde{\mathbf{n}}) \equiv \frac{2\pi}{L}\tilde{\mathbf{n}}$$

$$\rightarrow \quad (k_1, k_2, k_3) = \frac{2\pi}{L}(\tilde{n}_1, \tilde{n}_2, \tilde{n}_3)$$

2D Slice



Fourier Lattice:

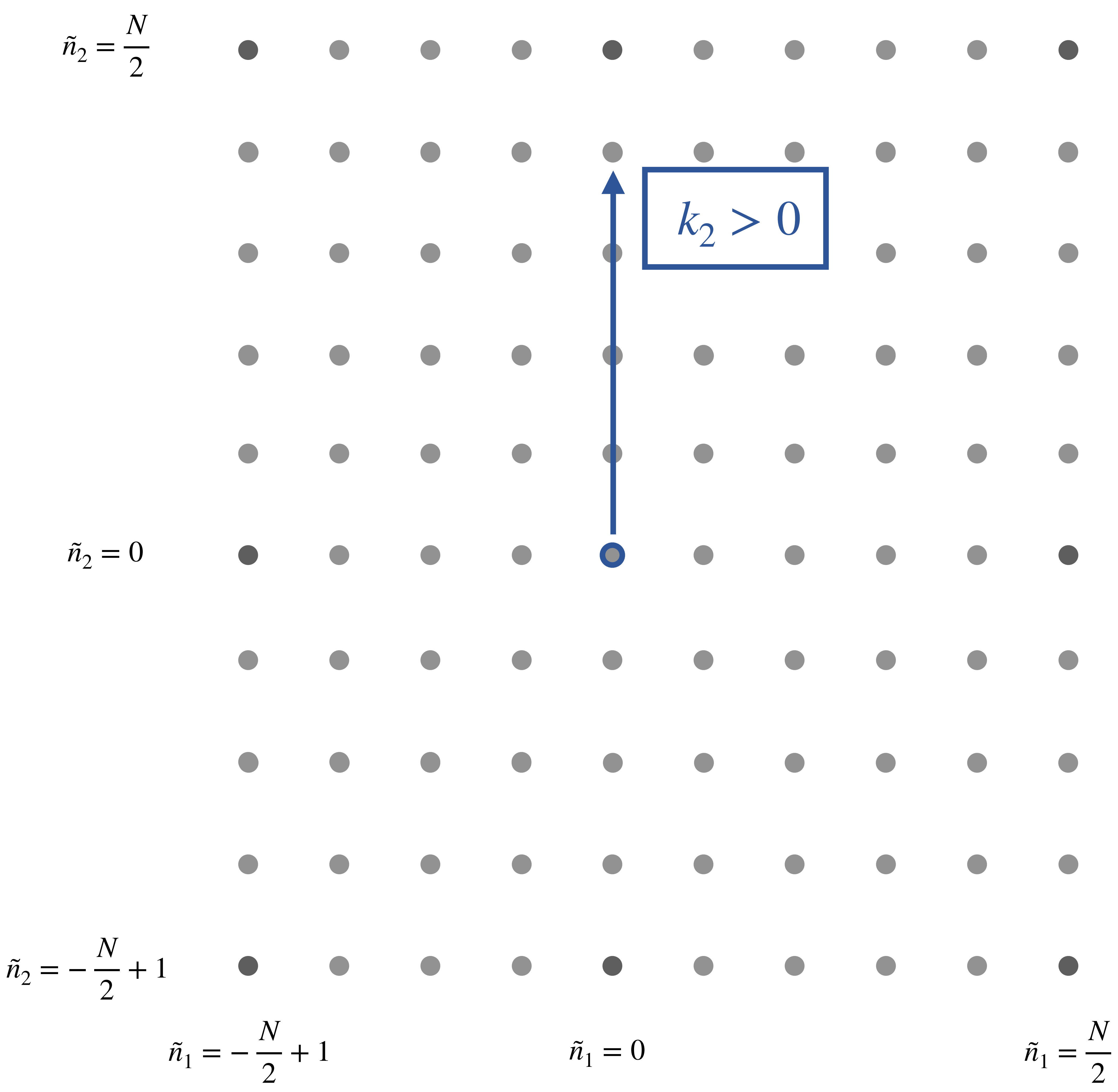
$$\tilde{n}_i = -\frac{N}{2} + 1, \ \dots \ , -1, 0, 1, \ \dots \ , \frac{N}{2}$$

FL captures **physical momenta:**

$$\mathbf{k}(\tilde{\mathbf{n}}) \equiv \frac{2\pi}{L}\tilde{\mathbf{n}}$$

$$\rightarrow \quad (k_1, k_2, k_3) = \frac{2\pi}{L}(\tilde{n}_1, \tilde{n}_2, \tilde{n}_3)$$

2D Slice



Fourier Lattice:

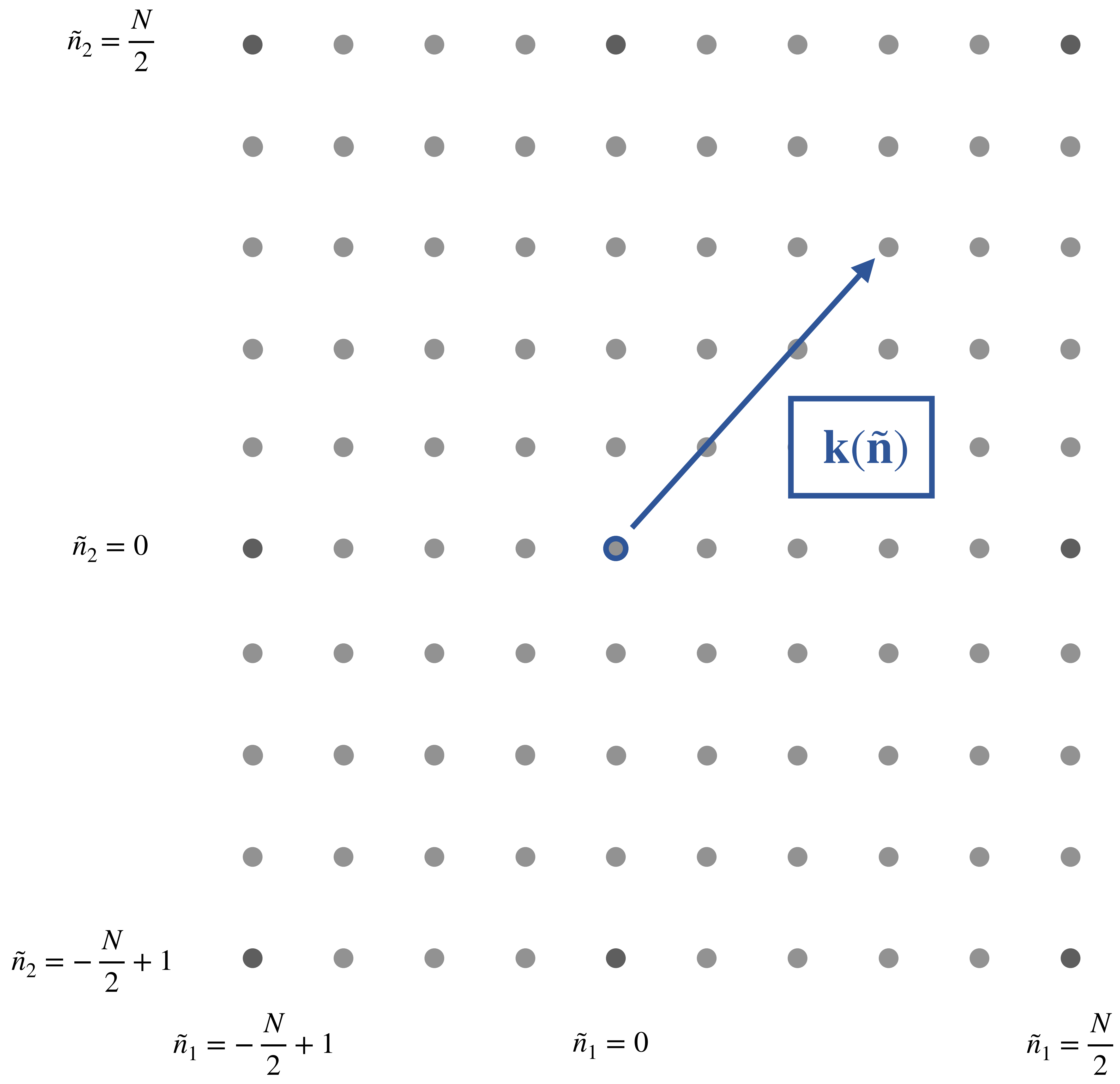
$$\tilde{n}_i = -\frac{N}{2}+1, \ \dots \ , -1, 0, 1, \ \dots \ , \frac{N}{2}$$

FL captures **physical momenta:**

$$\mathbf{k}(\tilde{\mathbf{n}}) \equiv \frac{2\pi}{L}\tilde{\mathbf{n}}$$

$$\rightarrow \quad (k_1, k_2, k_3) = \frac{2\pi}{L}(\tilde{n}_1, \tilde{n}_2, \tilde{n}_3)$$

2D Slice



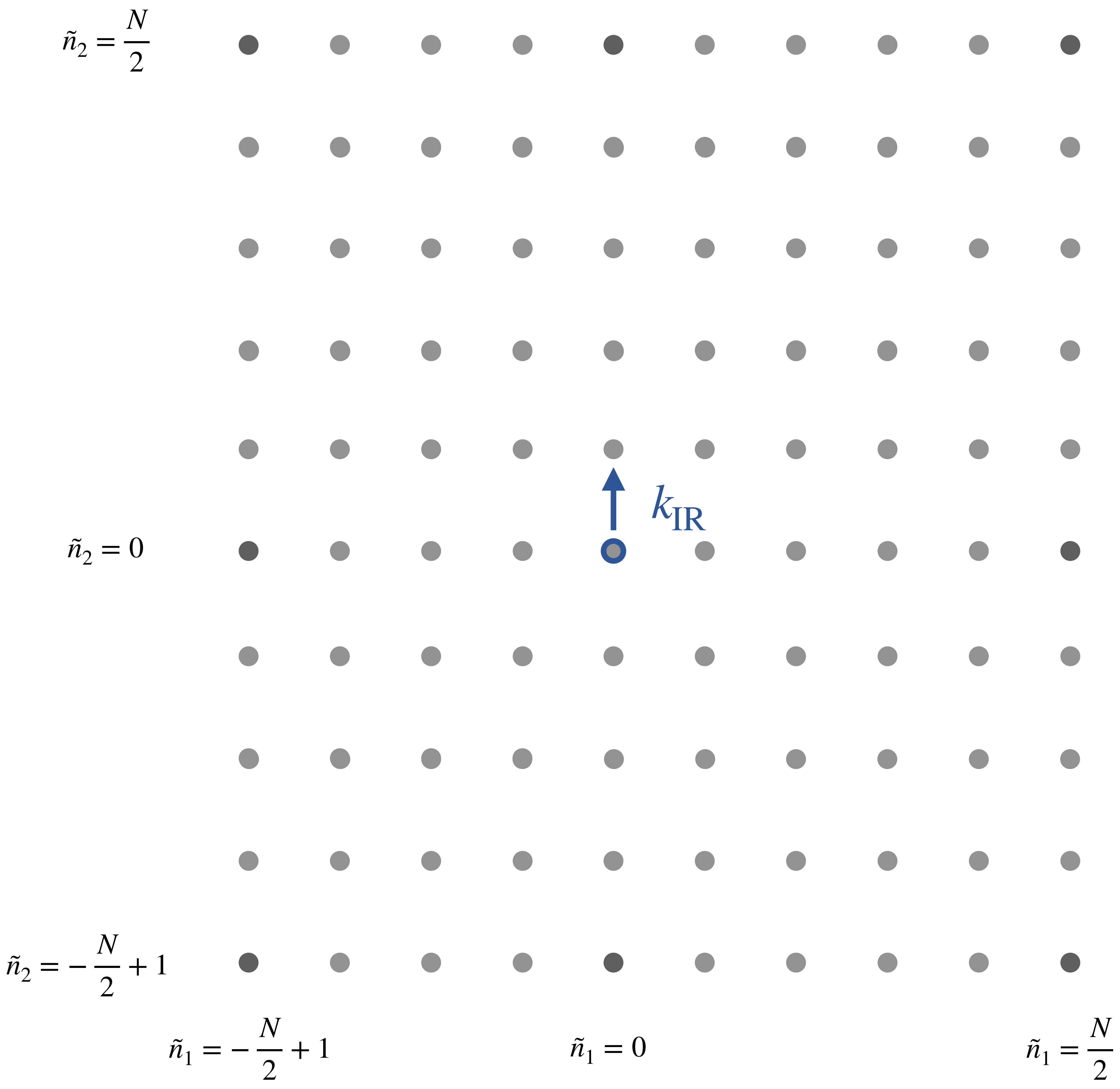
Fourier Lattice:

$$\tilde{n}_i = -\frac{N}{2} + 1, \ \dots \ , -1, 0, 1, \ \dots \ , \frac{N}{2}$$

Infrared Momentum: smallest momentum captured

$$k_{\rm IR} \equiv \frac{2\pi}{L}$$

2D Slice



Fourier Lattice:

$$\tilde{n}_i = -\frac{N}{2}+1,\ \dots\ ,\ -1, 0, 1,\ \dots\ ,\frac{N}{2}$$

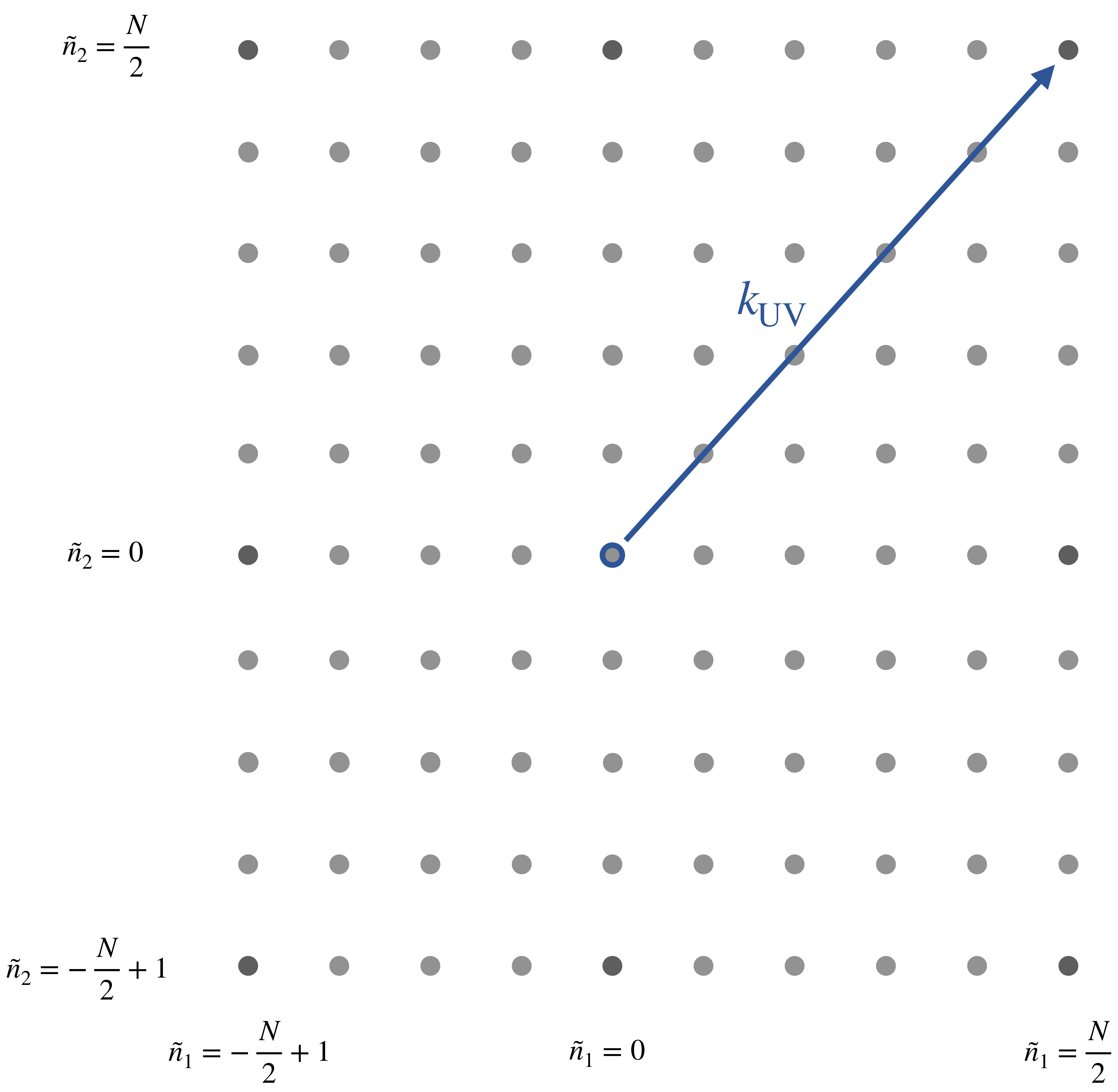
Infrared Momentum: smallest momentum captured

$$k_{\rm IR} \equiv \frac{2\pi}{L}$$

Ultraviolet Momentum: largest momentum captured

$$k_{\rm UV} \equiv \sqrt{d}\frac{N}{2}k_{\rm IR}$$

2D Slice



2. Fourier Lattice

Fourier Lattice:

$$\tilde{n}_i = -\frac{N}{2} + 1, \dots, -1, 0, 1, \dots, \frac{N}{2}$$

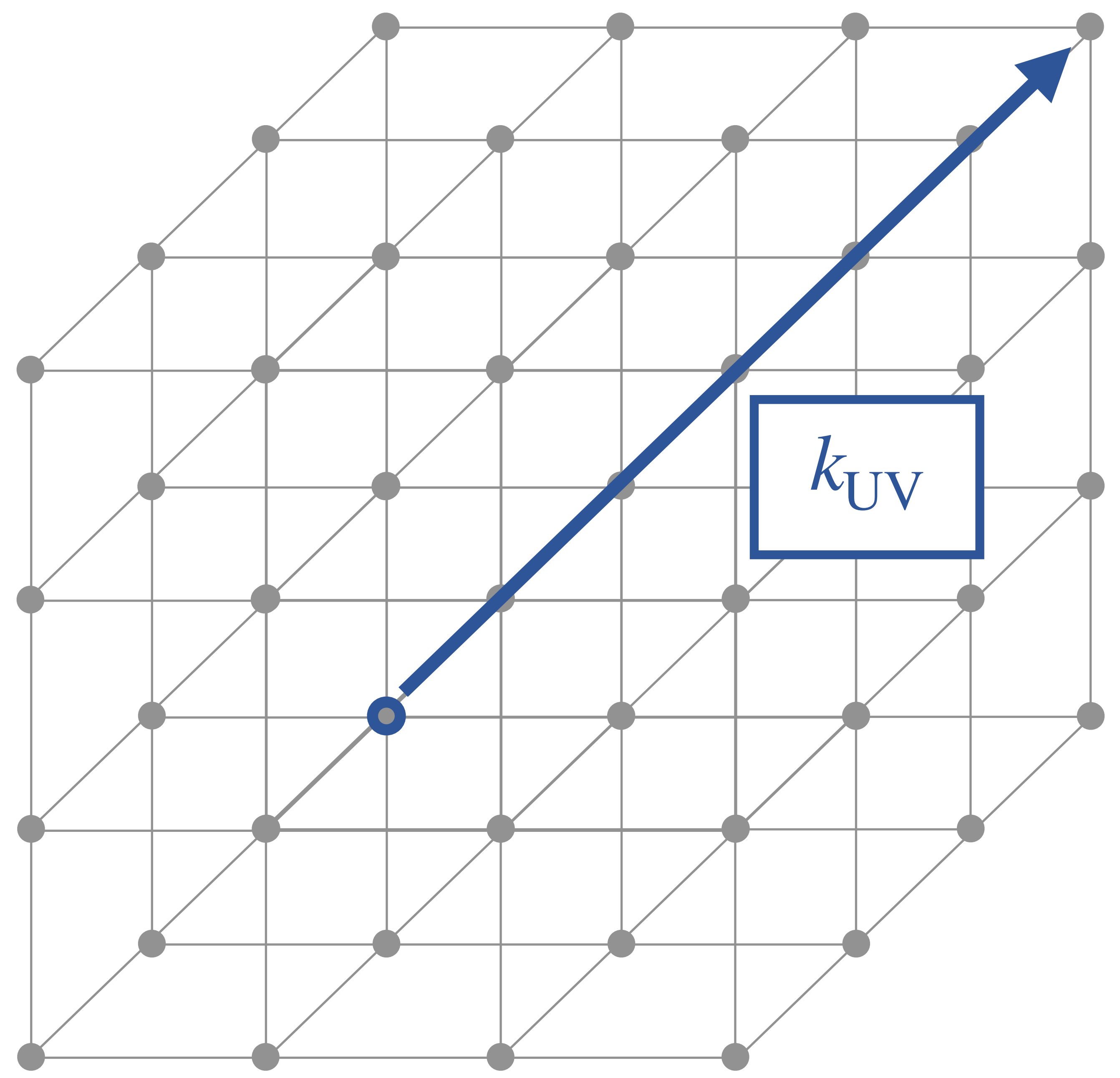
Infrared Momentum: smallest momentum captured

$$k_{\rm IR} \equiv \frac{2\pi}{L}$$

Ultraviolet Momentum: largest momentum captured

$$k_{\rm UV} \equiv \sqrt{d}\frac{N}{2}k_{\rm IR}$$

3D Lattice



2. Fourier Lattice

Real Lattice

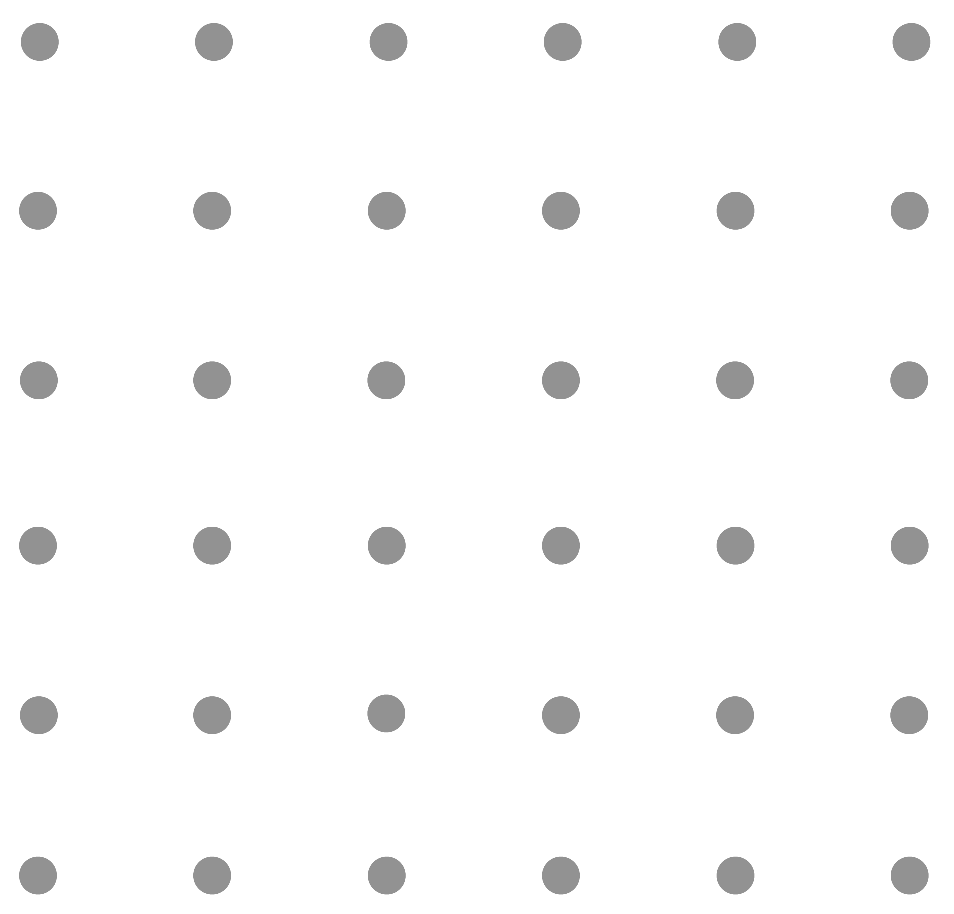
$$n_i = 1, \ \ldots, N \quad (i = 1, \ \ldots, d)$$

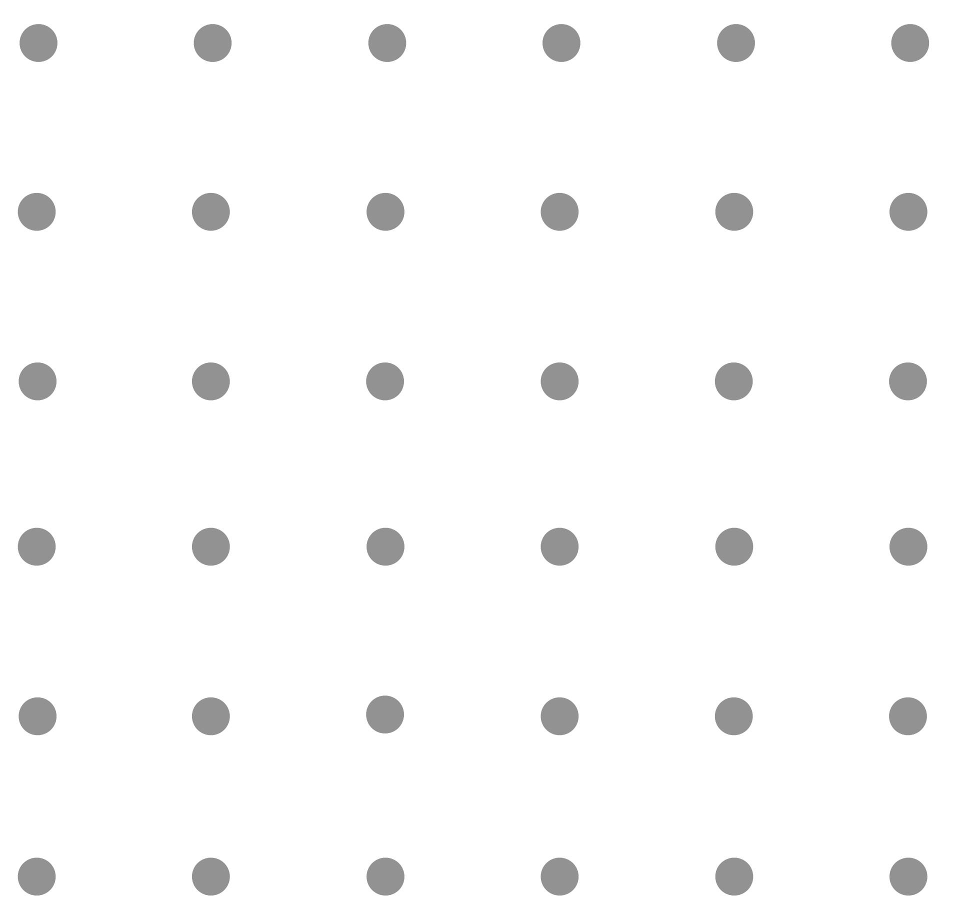
$$f(\mathbf{n} + N\hat{\imath}) \equiv f(\mathbf{n})$$

Fourier Lattice

$$\tilde{n}_i = -\frac{N}{2} + 1, \ \ldots, \ -1, 0, 1, \ \ldots, \frac{N}{2}$$

$$f(\tilde{\mathbf{n}} + N\hat{\imath}) \equiv f(\tilde{\mathbf{n}}) \qquad (i = 1, \ \ldots, d)$$





2. Fourier Lattice

Real Lattice

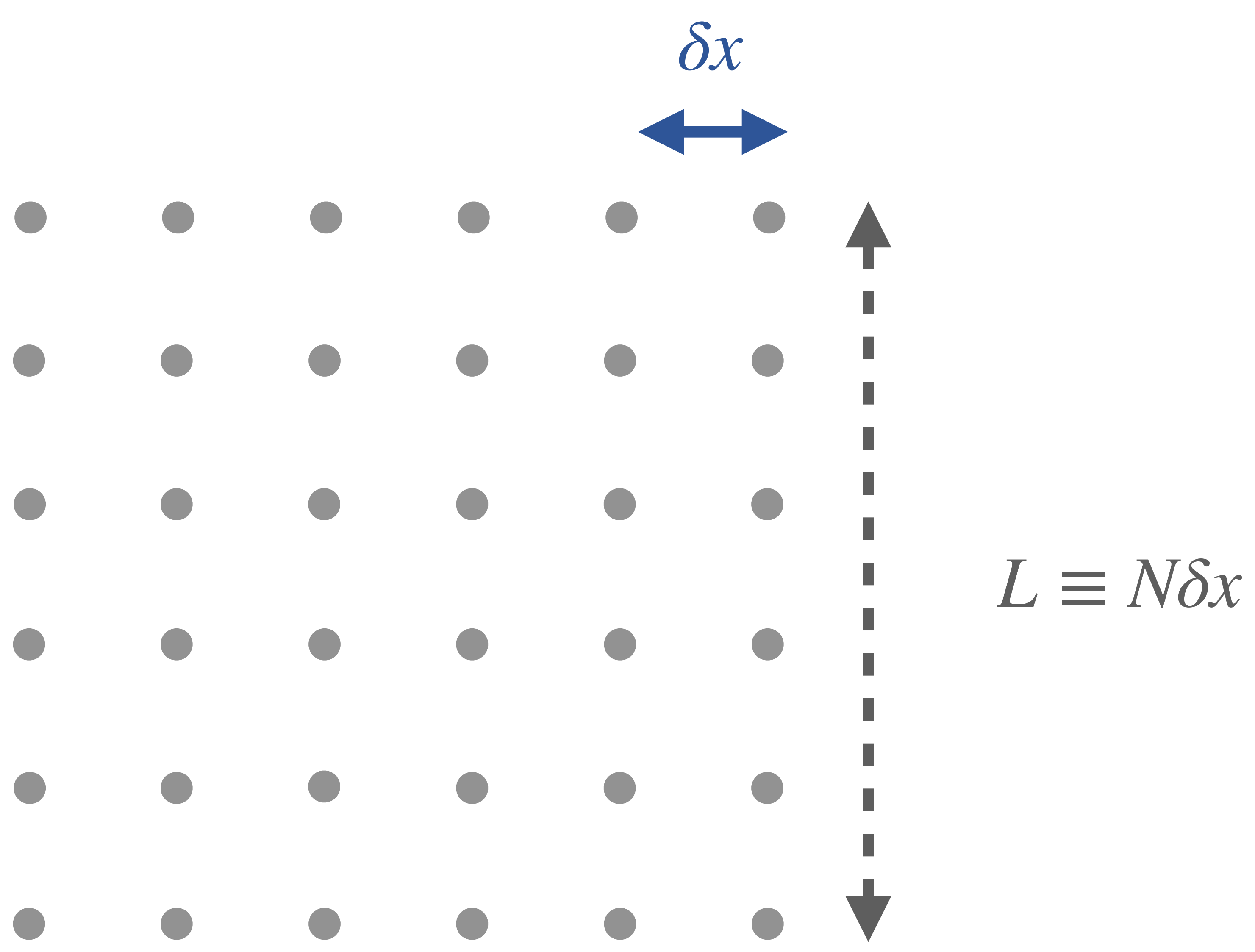
$$n_i = 1, \ \ldots, N \quad (i = 1, \ \ldots, d)$$

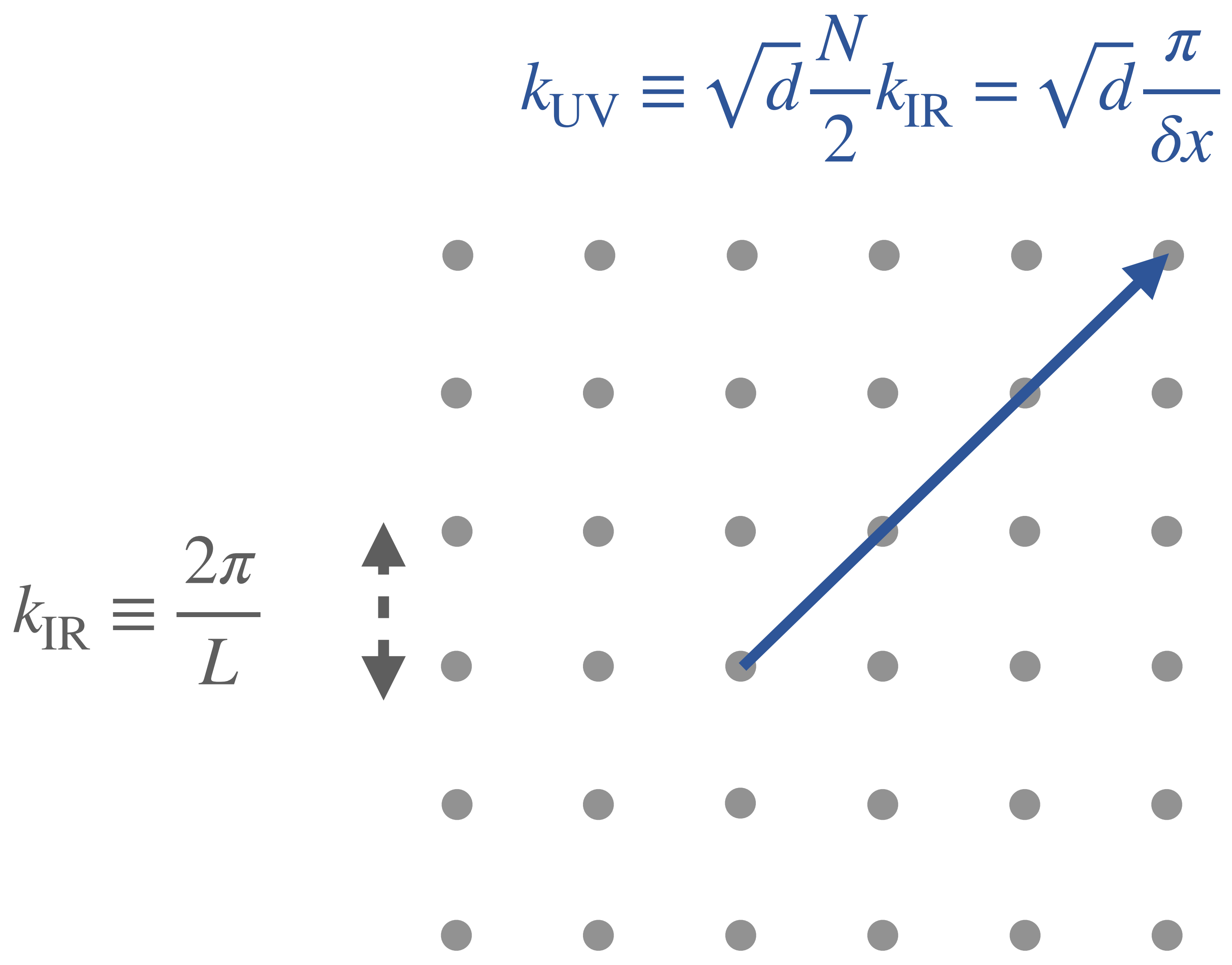
$$f(\mathbf{n} + N\hat{\imath}) \equiv f(\mathbf{n})$$

Fourier Lattice

$$\tilde{n}_i = -\frac{N}{2} + 1, \ \ldots, \ -1, 0, 1, \ \ldots, \frac{N}{2}$$

$$f(\tilde{\mathbf{n}} + N\hat{\imath}) \equiv f(\tilde{\mathbf{n}}) \qquad (i = 1, \ \ldots, d)$$





Real Lattice

$$n_i = 1, \ \dots, N \quad (i = 1, \ \dots, d)$$

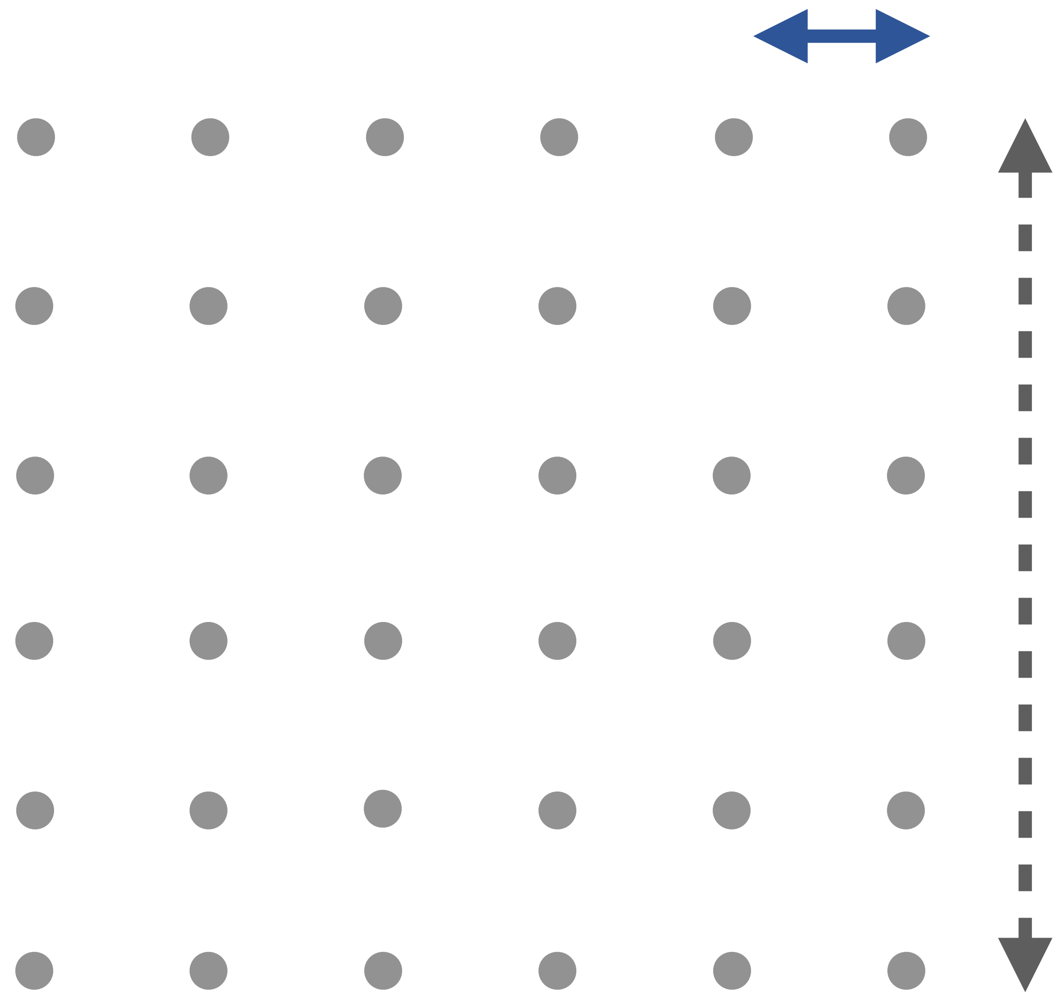
$$f(\mathbf{n} + N\hat{\imath}) \equiv f(\mathbf{n})$$

Fourier Lattice

$$\tilde{n}_i = -\frac{N}{2} + 1, \ \dots, \ -1, 0, 1, \ \dots, \frac{N}{2}$$

$$f(\tilde{\mathbf{n}} + N\hat{\imath}) \equiv f(\tilde{\mathbf{n}}) \qquad (i = 1, \ \dots, d)$$

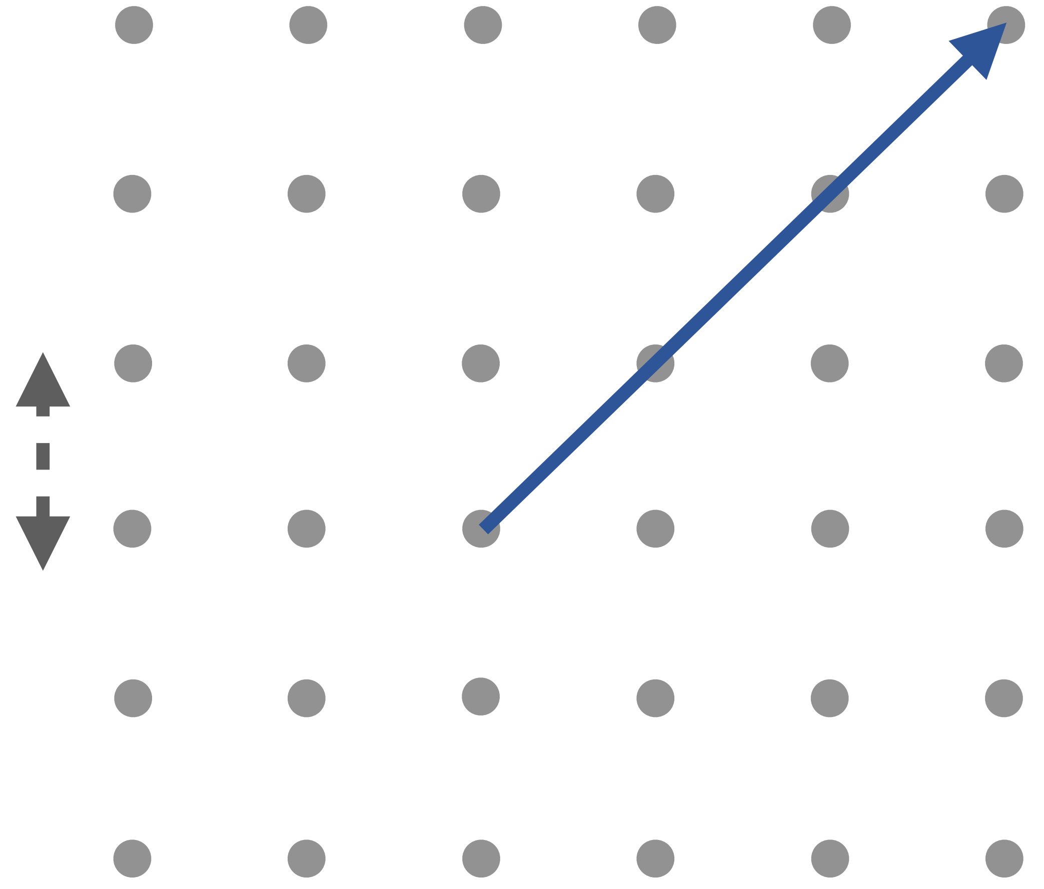
δx



$L \equiv N\delta x$

(reciprocal)

$$k_{\mathrm{UV}} \equiv \sqrt{d}\,\frac{N}{2}k_{\mathrm{IR}} = \sqrt{d}\,\frac{\pi}{\delta x}$$



$$k_{\mathrm{IR}} \equiv \frac{2\pi}{L}$$

4. Lattice Derivatives

4. Lattice Derivatives

Neutral derivative:

$$\boxed{[\nabla_i^{(0)} f](\mathbf{n}) = \frac{f(\mathbf{n}+\hat{\imath}) - f(\mathbf{n}-\hat{\imath})}{2\delta x^\mu}}$$

4. Lattice Derivatives

Neutral derivative:

displacement by one unit

$$\boxed{[\nabla_i^{(0)} f](\mathbf{n}) = \frac{f(\mathbf{n}+\hat{\imath}) - f(\mathbf{n}-\hat{\imath})}{2\delta x^\mu}}$$

lattice spacing

Neutral derivative:

$$\boxed{[\nabla_i^{(0)} f](\mathbf{n}) = \frac{f(\mathbf{n}+\hat{\imath}) - f(\mathbf{n}-\hat{\imath})}{2\delta x^{\mu}}} \longrightarrow \partial_i \mathtt{f}(x)\,|_{x\equiv\mathbf{n}\delta x} + \mathcal{O}(\delta x_i^2)$$

(recovers the continuum up to order δx_i^2)

4. Lattice Derivatives

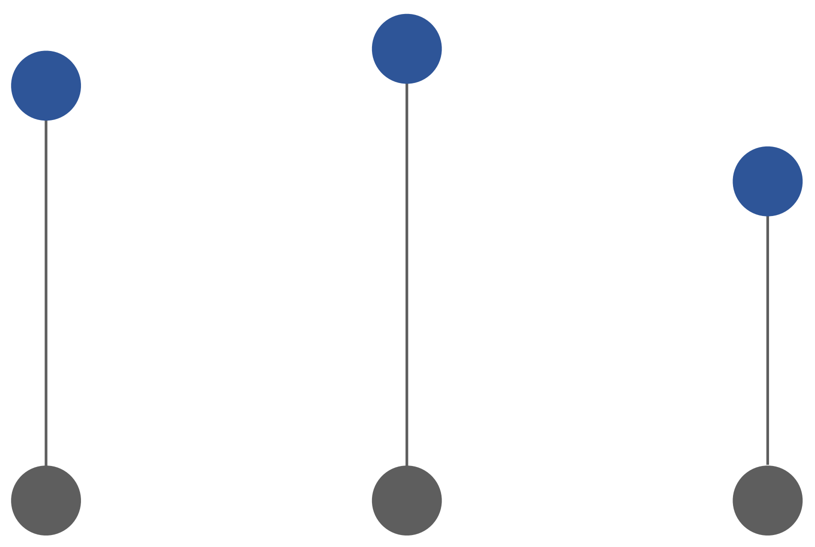
Neutral derivative:

$$\boxed{[\nabla_i^{(0)} f](\mathbf{n}) = \frac{f(\mathbf{n}+\hat{\imath}) - f(\mathbf{n}-\hat{\imath})}{2\delta x^\mu}} \quad \longrightarrow \quad \partial_i \mathtt{f}(x)\,|_{x\equiv\mathbf{n}\delta x} + \mathcal{O}(\delta x_i^2)$$

(recovers the continuum up to order δx_i^2)

Example:

$f(3)$



$n = 3$

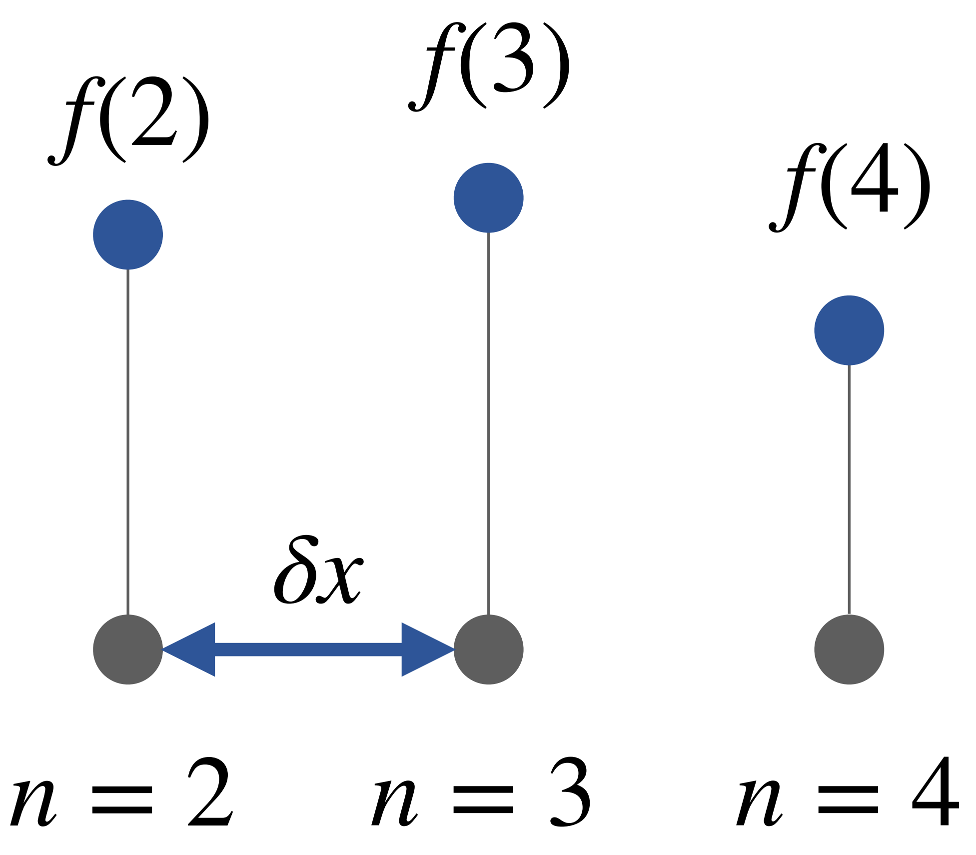
$$[\nabla_x^{(0)} f](3)$$

Neutral derivative:

$$\boxed{[\nabla_i^{(0)} f](\mathbf{n}) = \frac{f(\mathbf{n}+\hat{\imath}) - f(\mathbf{n}-\hat{\imath})}{2\delta x^\mu}} \quad \longrightarrow \quad \partial_i \mathtt{f}(x)\,|_{x\equiv \mathbf{n}\delta x} + \mathcal{O}(\delta x_i^2)$$

(recovers the continuum up to order δx_i^2)

Example:



$$[\nabla_x^{(0)} f](3) = \frac{f(4) - f(2)}{2\delta x} \quad \longrightarrow \quad \partial_x \mathtt{f}(3) + \mathcal{O}(\delta x^2)$$

4. Lattice Derivatives

Charged derivatives: forward: (+) backward: (-)

$$\boxed{[\nabla_i^{(\pm)} f](\mathbf{n}) = \frac{\pm f(\mathbf{n} \pm \hat{\imath}) \mp f(\mathbf{n})}{\delta x}}$$

4. Lattice Derivatives

Charged derivatives: forward: (+) backward: (-)

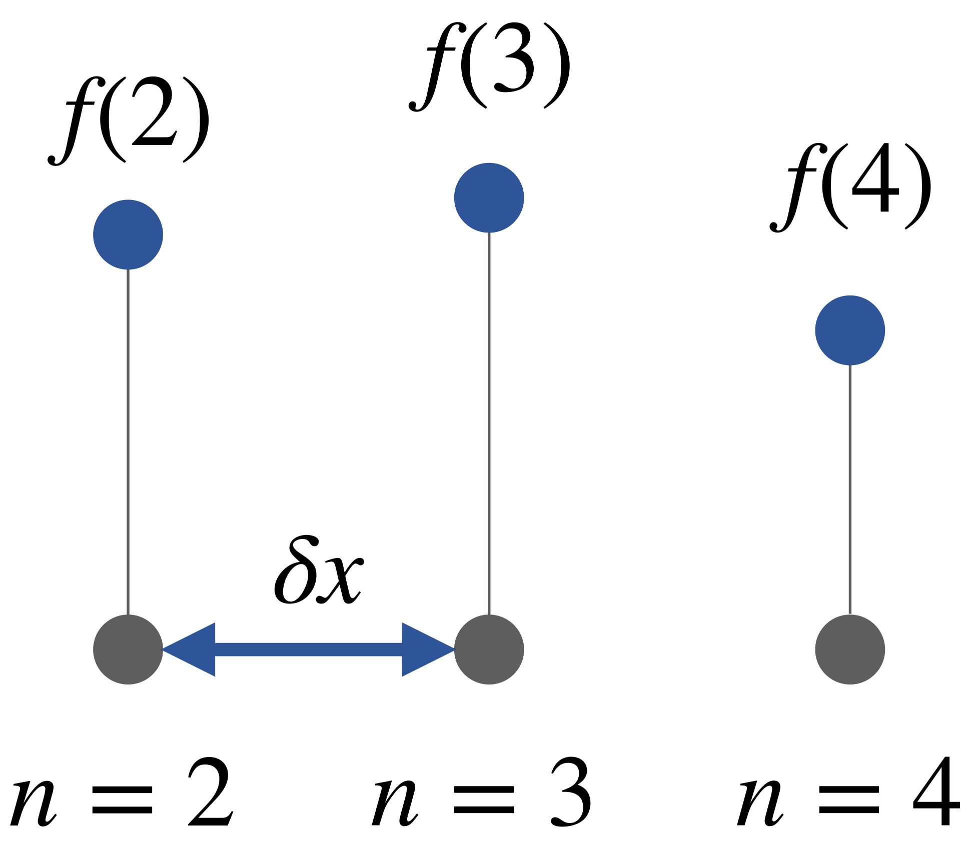
$$\boxed{[\nabla_i^{(\pm)} f](\mathbf{n}) = \frac{\pm f(\mathbf{n} \pm \hat{\imath}) \mp f(\mathbf{n})}{\delta x}} \longrightarrow \begin{cases} \partial_i \mathtt{f}(x)\,|_{x \equiv \mathbf{n}\delta x} + \mathcal{O}(\delta x_i) \\ \\ \partial_i \mathtt{f}(x)\,|_{x \equiv (\mathbf{n} \pm \hat{\imath}/2)\delta x} + \mathcal{O}(\delta x_i^2) \end{cases}$$

4. Lattice Derivatives

Charged derivatives: forward: (+) backward: (-)

$$\boxed{[\nabla_i^{(\pm)} f](\mathbf{n}) = \frac{\pm f(\mathbf{n} \pm \hat{\imath}) \mp f(\mathbf{n})}{\delta x}} \longrightarrow \begin{cases} \partial_i \mathtt{f}(x)\,|_{x \equiv \mathbf{n}\delta x} + \mathcal{O}(\delta x_i) \\ \\ \partial_i \mathtt{f}(x)\,|_{x \equiv (\mathbf{n} \pm \hat{\imath}/2)\delta x} + \mathcal{O}(\delta x_i^2) \end{cases}$$

Example:



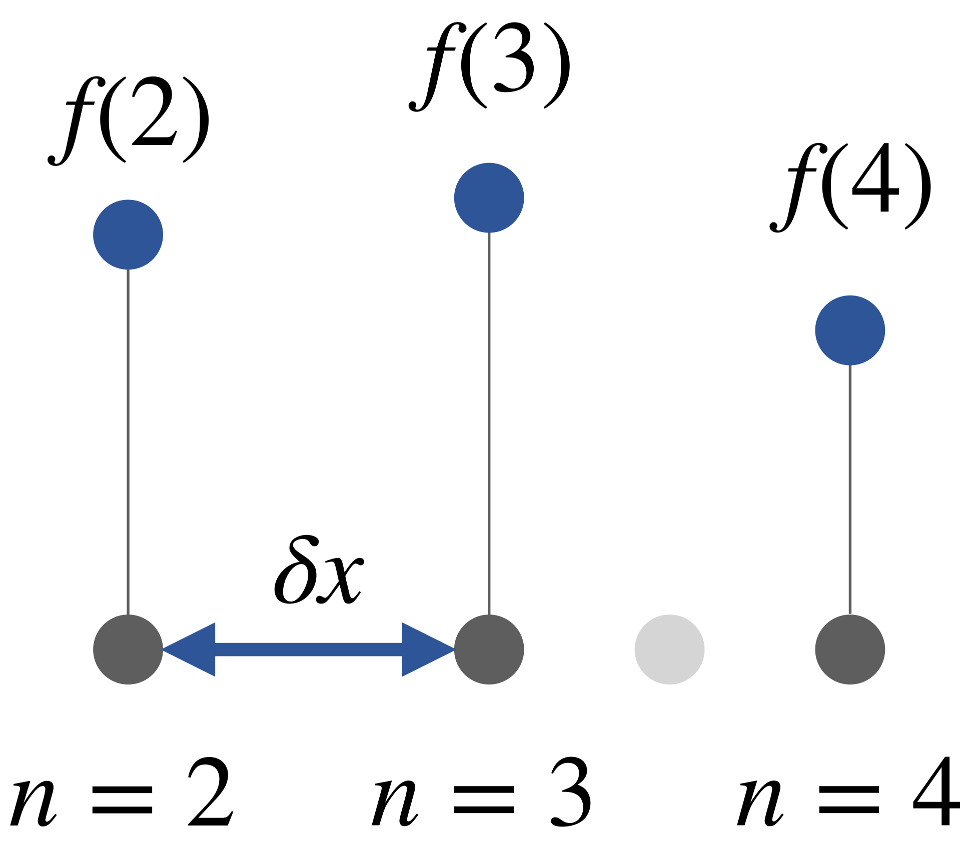
$$[\nabla_x^{(+)} f](3.0) = \frac{f(4) - f(3)}{\delta x} \longrightarrow \partial_x \mathtt{f}(3) + \mathcal{O}(\delta x)$$

4. Lattice Derivatives

Charged derivatives: forward: (+) backward: (-)

$$\boxed{[\nabla_i^{(\pm)} f](\mathbf{n}) = \frac{\pm f(\mathbf{n} \pm \hat{\imath}) \mp f(\mathbf{n})}{\delta x}} \longrightarrow \begin{cases} \partial_i \mathtt{f}(x)\,|_{x \equiv \mathbf{n}\delta x} + \mathcal{O}(\delta x_i) \\ \\ \partial_i \mathtt{f}(x)\,|_{x \equiv (\mathbf{n} \pm \hat{\imath}/2)\delta x} + \mathcal{O}(\delta x_i^2) \end{cases}$$

Example:



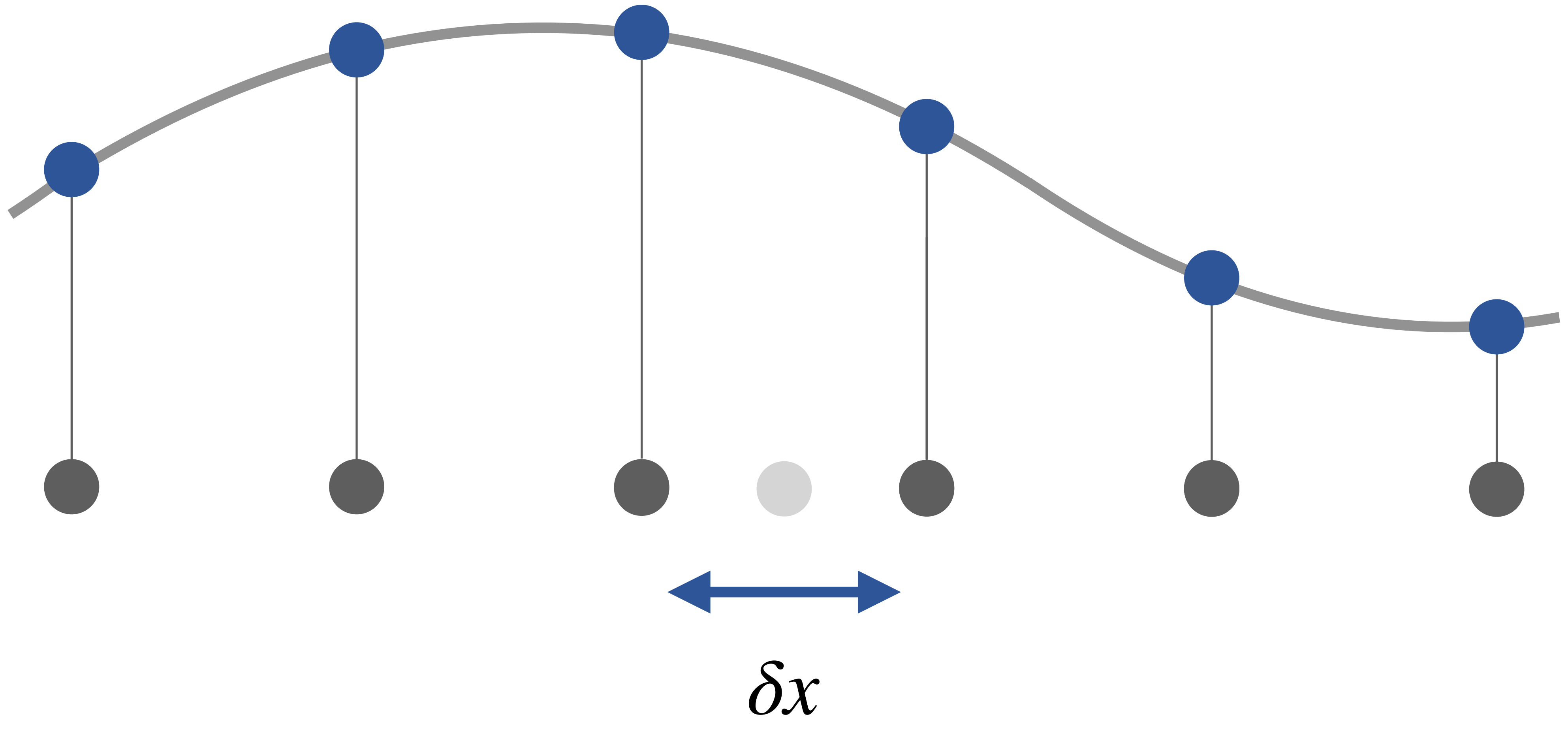
$$[\nabla_x^{(+)} f](3.0) = \frac{f(4) - f(3)}{\delta x} \longrightarrow \partial_x \mathtt{f}(3) + \mathcal{O}(\delta x)$$

$$[\nabla_x^{(+)} f](3.5) = \frac{f(4) - f(3)}{\delta x} \longrightarrow \partial_x \mathtt{f}(3.5) + \mathcal{O}(\delta x^2)$$

4. Lattice Derivatives

How well is a continuum function represented on the lattice?

- smallness of δx
- lattice derivative $\nabla_\mu^{(\mathrm{L})}$
- location where operator 'lives' $n_\mu \pm a_\mu$



5. Power Spectrum

5. Power Spectrum

Power spectrum in the continuum:

$$\langle \mathtt{f}^2(\mathbf{x},t)\rangle = \int d\log k\,\Delta_{\mathtt{f}}(k,t)\ ,\qquad \Delta_{\mathtt{f}}(k,t) = \frac{k^3}{2\pi^2}\mathscr{P}_{\mathtt{f}}(k,t)\ ,\qquad \langle \mathtt{f}_{\mathbf{k}}\mathtt{f}_{\mathbf{k}'}\rangle = (2\pi)^3\mathscr{P}_{\mathtt{f}}(k,t)\delta(\mathbf{k}-\mathbf{k}')$$

ensemble average

power spectrum

2-point function

5. Power Spectrum

Power spectrum in the continuum:

$$\langle \mathtt{f}^2(\mathbf{x},t)\rangle = \int d\log k\,\Delta_{\mathtt{f}}(k,t)\ ,\qquad \Delta_{\mathtt{f}}(k,t) = \frac{k^3}{2\pi^2}\mathscr{P}_{\mathtt{f}}(k,t)\ ,\qquad \langle \mathtt{f}_{\mathbf{k}}\mathtt{f}_{\mathbf{k}'}\rangle = (2\pi)^3\mathscr{P}_{\mathtt{f}}(k,t)\delta(\mathbf{k}-\mathbf{k}')$$

ensemble average

power spectrum

2-point function

Power spectrum on the lattice: (see [2006.15122])

$$\langle f^2\rangle_V = \sum_{|\tilde{\mathbf{n}}|}\Delta\log k(\tilde{\mathbf{n}})\Delta_f^{(\mathrm{L})}(|\tilde{\mathbf{n}}|)$$

(summation over bins)

Lattice Power Spectrum

$$\Delta_f^{(\mathrm{L})}(|\tilde{\mathbf{n}}|) \equiv \frac{k^3(\tilde{\mathbf{n}})}{2\pi^2}\frac{L^3}{N^6}\langle |f(\tilde{\mathbf{n}})|^2\rangle_{R(\tilde{\mathbf{n}})}$$

angular average over spherical shells

II. Scalar Field Dynamics on the Lattice

6. Discretising the Equations of Motion

6. Discretising the Equations of Motion

Action of a real scalar field in FLRW background, with $g_{\mu\nu} = \text{diag}(-a^{2\alpha}, a^2, a^2, a^2)$:

$$S = -\int d^4x \sqrt{-g}\left(\frac{1}{2}\partial_\mu\phi\partial^\mu\phi + V(\phi)\right)$$

6. Discretising the Equations of Motion

Action of a real scalar field in FLRW background, with $g_{\mu\nu} = \mathrm{diag}(-a^{2\alpha}, a^2, a^2, a^2)$:

$$S = -\int d^4x\sqrt{-g}\left(\frac{1}{2}\partial_\mu\phi\partial^\mu\phi + V(\phi)\right)$$

in terms of dimensionless variables:

$$d\tilde{\eta} \equiv a^{-\alpha}\omega_* dt\,, \quad d\tilde{x} \equiv \omega_* dx\,, \quad \tilde{\phi} \equiv \frac{\phi}{f_*}\,, \quad \text{with} \quad \tilde{V}(\tilde{\phi}) \equiv \left.\frac{V(\phi)}{f_*^2\omega_*^2}\right|_{\phi=f_*\tilde{\phi}}$$

Action of a real scalar field in FLRW background, with $g_{\mu\nu} = \mathrm{diag}(-a^{2\alpha}, a^2, a^2, a^2)$:

$$S = -\int d^4x\sqrt{-g}\left(\frac{1}{2}\partial_\mu\phi\partial^\mu\phi + V(\phi)\right)$$

in terms of dimensionless variables:

$$d\tilde{\eta} \equiv a^{-\alpha}\omega_* dt\,,\quad d\tilde{x} \equiv \omega_* dx\,,\quad \tilde{\phi} \equiv \frac{\phi}{f_*}\,,\quad \text{with}\quad \tilde{V}(\tilde{\phi}) \equiv \left.\frac{V(\phi)}{f_*^2\omega_*^2}\right|_{\phi=f_*\tilde{\phi}}$$

Action in program variables:

$$\tilde{S} = -\left(\frac{f_*}{\omega_*}\right)^2\int d^3\tilde{x}d\tilde{\eta}\left(\frac{1}{2}a^{3-\alpha}(\tilde{\phi}')^2 - \frac{1}{2}a^{1+\alpha}(\tilde{\nabla}_i\tilde{\phi})^2 - a^{3+\alpha}\tilde{V}(\tilde{\phi})\right)$$

6. Discretising the Equations of Motion

Action of a real scalar field in FLRW background, with $g_{\mu\nu} = \mathrm{diag}(-a^{2\alpha}, a^2, a^2, a^2)$:

$$S = -\int d^4x\sqrt{-g}\left(\frac{1}{2}\partial_\mu\phi\partial^\mu\phi + V(\phi)\right)$$

in terms of dimensionless variables:

$$d\tilde{\eta} \equiv a^{-\alpha}\omega_* dt\,, \quad d\tilde{x} \equiv \omega_* dx\,, \quad \tilde{\phi} \equiv \frac{\phi}{f_*}\,, \quad \text{with} \quad \tilde{V}(\tilde{\phi}) \equiv \left.\frac{V(\phi)}{f_*^2\omega_*^2}\right|_{\phi=f_*\tilde{\phi}}$$

Action in program variables:

$$\tilde{S} = -\left(\frac{f_*}{\omega_*}\right)^2\int d^3\tilde{x}d\tilde{\eta}\left(\frac{1}{2}a^{3-\alpha}(\tilde{\phi}')^2 - \frac{1}{2}a^{1+\alpha}(\tilde{\nabla}_i\tilde{\phi})^2 - a^{3+\alpha}\tilde{V}(\tilde{\phi})\right)$$

→ $\delta S_\phi = 0$ to obtain equation of motion

The continuum equations of motion are:

- Equation of motion of a real scalar field:

$$\tilde{\phi}'' - a^{-2(1-\alpha)}\,\tilde{\nabla}^2\tilde{\phi} + (3-\alpha)\frac{a'}{a}\tilde{\phi}' + a^{2\alpha}\tilde{V}_{,\tilde{\phi}} = 0$$

6. Discretising the Equations of Motion

The continuum equations of motion are:

- Equation of motion of a real scalar field:

$$\tilde{\phi}'' - a^{-2(1-\alpha)}\,\tilde{\nabla}^2\tilde{\phi} + (3-\alpha)\frac{a'}{a}\tilde{\phi}' + a^{2\alpha}\tilde{V}_{,\tilde{\phi}} = 0$$

- Equation of motion of expanding background (Friedmann equations):

$$a'' = \frac{a^{2\alpha+1}}{3}\left(\frac{f_*}{m_p}\right)^2\left[(\alpha-2)\tilde{E}_K + \alpha\tilde{E}_G + (\alpha+1)\tilde{E}_V\right]\ , \qquad (a')^2 = \frac{a^{2\alpha+2}}{3}\left(\frac{f_*}{m_p}\right)^2\left[\tilde{E}_K + \tilde{E}_G + \tilde{E}_V\right]$$

6. Discretising the Equations of Motion

The continuum equations of motion are:

- Equation of motion of a real scalar field:

$$\tilde{\phi}'' - a^{-2(1-\alpha)}\,\tilde{\nabla}^2\tilde{\phi} + (3-\alpha)\frac{a'}{a}\tilde{\phi}' + a^{2\alpha}\tilde{V}_{,\tilde{\phi}} = 0$$

- Equation of motion of expanding background (Friedmann equations):

$$a'' = \frac{a^{2\alpha+1}}{3}\left(\frac{f_*}{m_p}\right)^2\left[(\alpha-2)\tilde{E}_K + \alpha\tilde{E}_G + (\alpha+1)\tilde{E}_V\right]\ , \qquad (a')^2 = \frac{a^{2\alpha+2}}{3}\left(\frac{f_*}{m_p}\right)^2\left[\tilde{E}_K + \tilde{E}_G + \tilde{E}_V\right]$$

volume averaged kinetic, gradient and potential energy density:

$$\tilde{E}_K \equiv \frac{1}{2a^{2\alpha}}\left\langle(\tilde{\phi}')^2\right\rangle \qquad \tilde{E}_G \equiv \frac{1}{2a^2}\sum_i\left\langle(\nabla_i\tilde{\phi})^2\right\rangle \qquad \tilde{E}_V \equiv \left\langle V(\tilde{\phi})\right\rangle$$

6. Discretising the Equations of Motion

How do we discretise these equations of motion?

6. Discretising the Equations of Motion

How do we discretise these equations of motion?

$$\boxed{\partial_\mu \quad \longrightarrow \quad \nabla_\mu^{(\mathrm{L})}}$$

6. Discretising the Equations of Motion

How do we discretise these equations of motion?

$$\boxed{\partial_\mu \quad \longrightarrow \quad \nabla_\mu^{(\mathrm{L})}}$$

(works for scalar fields!)

6. Discretising the Equations of Motion

How do we discretise these equations of motion?

$$\boxed{\partial_\mu \quad \longrightarrow \quad \nabla_\mu^{(\mathrm{L})}}$$

(works for scalar fields!)

- Discretise the action:
  - ➔ see [2006.15122]
- Discretise the equation of motion:

$$\tilde{\phi}'' - a^{-2(1-\alpha)} \sum_i \; \tilde{\partial}_i \, \tilde{\partial}_i \, \tilde{\phi} \quad + (3-\alpha)\frac{a'}{a}\tilde{\phi}' + a^{3+\alpha}\tilde{V}_{,\tilde{\phi}} = 0$$

6. Discretising the Equations of Motion

How do we discretise these equations of motion?

$$\boxed{\partial_\mu \longrightarrow \nabla^{(\mathrm{L})}_\mu}$$

(works for scalar fields!)

- Discretise the action:
  - ➔ see [2006.15122]

- Discretise the equation of motion:

$$\tilde{\phi}'' - a^{-2(1-\alpha)} \sum_i \tilde{\partial}_i \tilde{\partial}_i \tilde{\phi} + (3-\alpha)\frac{a'}{a}\tilde{\phi}' + a^{3+\alpha}\tilde{V}_{,\tilde{\phi}} = 0 \quad , \qquad \tilde{E}_G \equiv \frac{1}{2a^2}\sum_i \left\langle (\tilde{\partial}_i \tilde{\phi})^2 \right\rangle$$

6. Discretising the Equations of Motion

How do we discretise these equations of motion?

$$\boxed{\partial_\mu \longrightarrow \nabla_\mu^{(\mathrm{L})}}$$

(works for scalar fields!)

- Discretise the action:
  - ➔ see [2006.15122]
- Discretise the equation of motion:

$$\tilde{\phi}'' - a^{-2(1-\alpha)} \sum_i \tilde{\nabla}_i^{(\mathrm{L})} \tilde{\nabla}_i^{(\mathrm{L})} \tilde{\phi} + (3-\alpha)\frac{a'}{a}\tilde{\phi}' + a^{3+\alpha}\tilde{V}_{,\tilde{\phi}} = 0 \quad , \qquad \tilde{E}_G \equiv \frac{1}{2a^2}\sum_i \left\langle (\nabla_i^{(\mathrm{L})}\tilde{\phi})^2 \right\rangle$$

How do we discretise these equations of motion?

$$\boxed{\partial_\mu \quad \longrightarrow \quad \nabla_\mu^{(\mathrm{L})}}$$

(works for scalar fields!)

- Discretise the action:
  - see [2006.15122]

- Discretise the equation of motion:

$$\tilde{\phi}'' - a^{-2(1-\alpha)} \sum_i \tilde{\nabla}_i^{(-)} \tilde{\nabla}_i^{(+)} \tilde{\phi} + (3-\alpha)\frac{a'}{a}\tilde{\phi}' + a^{3+\alpha}\tilde{V}_{,\tilde{\phi}} = 0 \quad , \qquad \tilde{E}_G \equiv \frac{1}{2a^2} \sum_i \left\langle (\nabla_i^{(+)}\tilde{\phi})^2 \right\rangle$$

  - accurate to $\mathcal{O}(dx_i^2)$ at lattice site $\mathbf{n}$.

7. Evolving the System of Equations

7. Evolution Algorithms

Which equations of motion are we solving for?

Which equations of motion are we solving for?

- Equation of motion of the scalar field
- Second Friedmann equation
  - ➔ Use the first Friedmann equation as a constraint equation to check energy conservation

7. Evolution Algorithms

Which equations of motion are we solving for?

- Equation of motion of the scalar field
- Second Friedmann equation
  - ➡ Use the first Friedmann equation as a constraint equation to check energy conservation

How do we evolve the equations of motion?

Which equations of motion are we solving for?

- Equation of motion of the scalar field
- Second Friedmann equation
  - ➡ Use the first Friedmann equation as a constraint equation to check energy conservation

How do we evolve the equations of motion?

- Evolve the field amplitudes $\{\tilde{\phi}, a\}$ and conjugate momenta $\{\tilde{\pi}, \pi_a\}$ at every lattice site
- Solve the equations of motion by applying some evolution algorithm

7. Evolution Algorithms

Evolution algorithms:

7. Evolution Algorithms

Evolution algorithms:

- Symplectic integrators

  → good for Hamiltonian systems, conserve energy

7. Evolution Algorithms

Evolution algorithms:

- Symplectic integrators

  → good for Hamiltonian systems, conserve energy

  examples: LeapFrog, Verlet methods, Yoshida, …

7. Evolution Algorithms

Evolution algorithms:

- Symplectic integrators

  ➔ good for Hamiltonian systems, conserve energy

  examples: LeapFrog, Verlet methods, Yoshida, …

- Non-symplectic integrators

  ➔ broader applicability, e.g. non-Hamiltonian, dissipative systems

7. Evolution Algorithms

Evolution algorithms:

- Symplectic integrators

  ➔ good for Hamiltonian systems, conserve energy

  examples: LeapFrog, Verlet methods, Yoshida, …

- Non-symplectic integrators

  ➔ broader applicability, e.g. non-Hamiltonian, dissipative systems

  examples: Runge-Kutta methods, …

7. Evolution Algorithms

Evolution algorithms:

- Symplectic integrators

  → good for Hamiltonian systems, conserve energy

  examples: LeapFrog, Verlet methods, Yoshida, …

- Non-symplectic integrators

  → broader applicability, e.g. non-Hamiltonian, dissipative systems

  examples: Runge-Kutta methods, …

Rewrite the equations of motion into a Hamiltonian like scheme:

Rewrite the equations of motion into a Hamiltonian like scheme:

$$\pi(\mathbf{x},\eta) = \mathscr{D}[f'(\mathbf{x},\eta), a(\eta), \pi_a(\eta)\}]$$

$$\pi'(\mathbf{x},\eta) = \mathscr{K}[f(\mathbf{x},\eta), a(\eta)\}]$$

$$\pi_a(\eta) = a'(\eta)$$

$$\pi_a'(\eta) = \mathscr{K}_a[a(\eta), E_V(\eta), E_K(\eta), E_G(\eta)]$$

Rewrite the equations of motion into a Hamiltonian like scheme:

$$\pi(\mathbf{x},\eta) = \mathscr{D}[f'(\mathbf{x},\eta), a(\eta), \pi_a(\eta)\}]$$

$$\pi'(\mathbf{x},\eta) = \mathscr{K}[f(\mathbf{x},\eta), a(\eta)\}]$$

$$\pi_a(\eta) = a'(\eta)$$

$$\pi_a'(\eta) = \mathscr{K}_a[a(\eta), E_V(\eta), E_K(\eta), E_G(\eta)]$$

apply the following trick:

$$\tilde{\phi}'' = a^{-2(1-\alpha)}\tilde{\nabla}^2\tilde{\phi} - (3-\alpha)\frac{a'}{a}\tilde{\phi}' - a^{3+\alpha}\tilde{V}_{,\tilde{\phi}}$$

Rewrite the equations of motion into a Hamiltonian like scheme:

$$\pi(\mathbf{x},\eta) = \mathscr{D}[f'(\mathbf{x},\eta), a(\eta), \pi_a(\eta)\}]$$

$$\pi'(\mathbf{x},\eta) = \mathscr{K}[f(\mathbf{x},\eta), a(\eta)\}]$$

$$\pi_a(\eta) = a'(\eta)$$

$$\pi_a'(\eta) = \mathscr{K}_a[a(\eta), E_V(\eta), E_K(\eta), E_G(\eta)]$$

apply the following trick:

$$\tilde{\phi}'' = a^{-2(1-\alpha)}\tilde{\nabla}^2\tilde{\phi} - (3-\alpha)\frac{a'}{a}\tilde{\phi}' - a^{3+\alpha}\tilde{V}_{,\tilde{\phi}} \quad \rightarrow \quad (\overbrace{a^{(3-\alpha)}\tilde{\phi}'}^{\equiv\tilde{\pi}})' = a^{1+\alpha}\tilde{\nabla}^2\tilde{\phi} - a^{3+\alpha}\tilde{V}_{,\tilde{\phi}}$$

Rewrite the equations of motion into a Hamiltonian like scheme:

$$\pi(\mathbf{x},\eta) = \mathscr{D}[f'(\mathbf{x},\eta), a(\eta), \pi_a(\eta)\}]$$

$$\pi'(\mathbf{x},\eta) = \mathscr{K}[f(\mathbf{x},\eta), a(\eta)\}]$$

$$\pi_a(\eta) = a'(\eta)$$

$$\pi_a'(\eta) = \mathscr{K}_a[a(\eta), E_V(\eta), E_K(\eta), E_G(\eta)]$$

$\Rightarrow$

$$\tilde{\pi} = a^{3-\alpha}\tilde{\phi}'$$

$$\tilde{\pi}' = a^{1+\alpha}\tilde{\nabla}^2\tilde{\phi} - a^{3+\alpha}\tilde{V}_{,\tilde{\phi}}$$

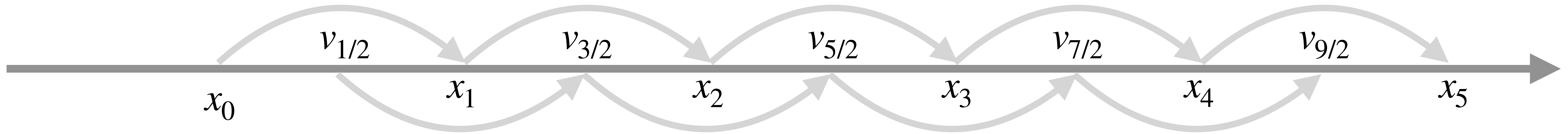
$$\pi_a \equiv b = a'$$

$$\pi_a' = \frac{a^{2\alpha+1}}{3}\left(\frac{f_*}{m_p}\right)^2\left[(\alpha-2)\tilde{E}_K + \alpha\tilde{E}_G + (\alpha+1)\tilde{E}_V\right]$$

apply the following trick:

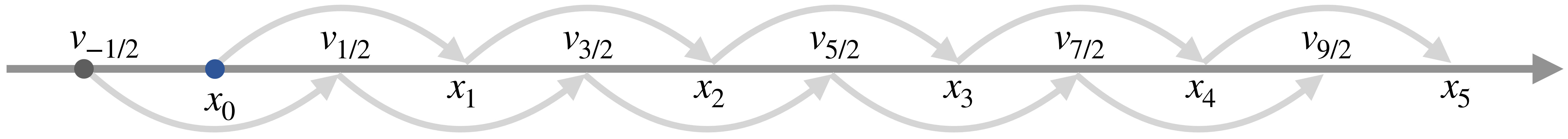
$$\tilde{\phi}'' = a^{-2(1-\alpha)}\tilde{\nabla}^2\tilde{\phi} - (3-\alpha)\frac{a'}{a}\tilde{\phi}' - a^{3+\alpha}\tilde{V}_{,\tilde{\phi}} \quad \Rightarrow \quad (\underbrace{a^{(3-\alpha)}\tilde{\phi}'}_{\equiv\,\tilde{\pi}})' = a^{1+\alpha}\tilde{\nabla}^2\tilde{\phi} - a^{3+\alpha}\tilde{V}_{,\tilde{\phi}}$$

LeapFrog:



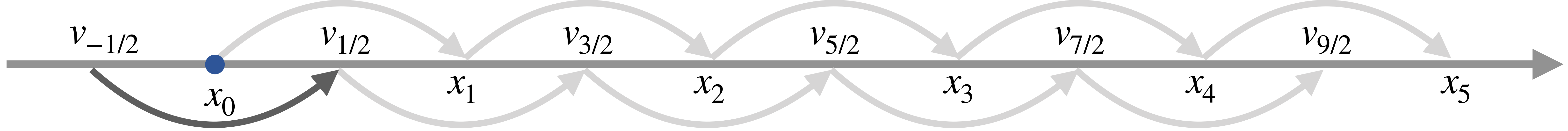
7. Evolution Algorithms: LeapFrog

LeapFrog:



$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

LeapFrog:

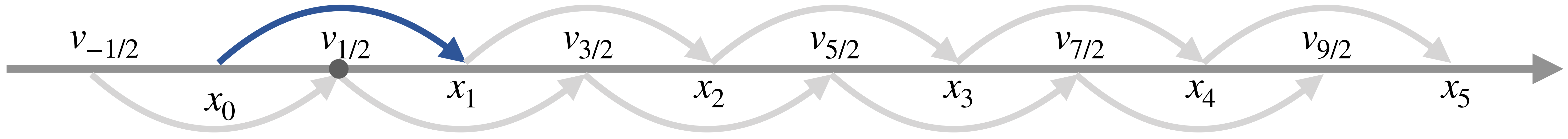


$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0 , \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta} , \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

7. Evolution Algorithms: LeapFrog

LeapFrog:



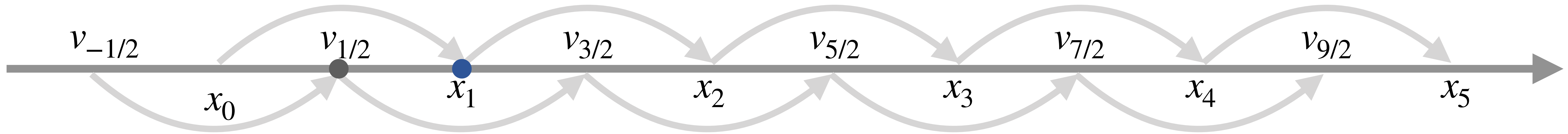
$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha+1)\tilde{E}_V \right] \delta\tilde{\eta}\,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

7. Evolution Algorithms: LeapFrog

LeapFrog:

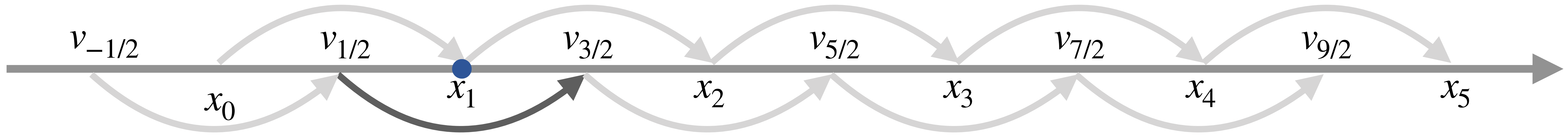


$$IC : \{\tilde\phi, a\} \text{ at } \tilde\eta_0\,, \quad \{\tilde\pi_{-0/2}, b_{-0/2}\} \text{ at } \tilde\eta_{-0/2} = \tilde\eta_0 - 0.5\delta\tilde\eta$$

$$\begin{cases} \tilde\pi_{+0/2} = \tilde\pi_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde\nabla_i^- \tilde\nabla_i^+ \tilde\phi_{+0} - a^{3+\alpha} \tilde V_{,\tilde\phi} \right) \delta\tilde\eta \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde E}_K + \alpha \tilde E_G + (\alpha+1)\tilde E_V \right] \delta\tilde\eta\,, \end{cases} \qquad \bar{\tilde E}_K \equiv (\tilde E_{K,-0/2} + \tilde E_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde\eta \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde\phi_{+0} = \tilde\phi + a_{+0/2}^{-(3-\alpha)} \tilde\pi_{+0/2} \delta\tilde\eta \end{cases}$$

LeapFrog:



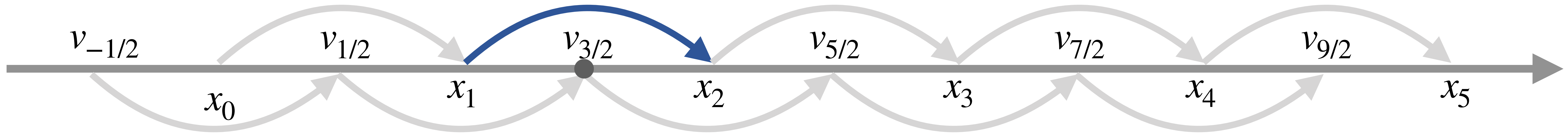
$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0 \,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta} \,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

7. Evolution Algorithms: LeapFrog

LeapFrog:

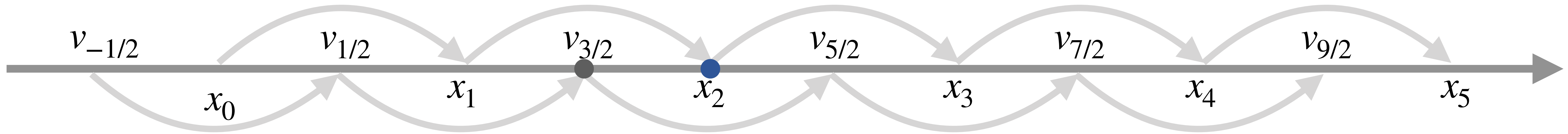


$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0 \,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta} \,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

LeapFrog:



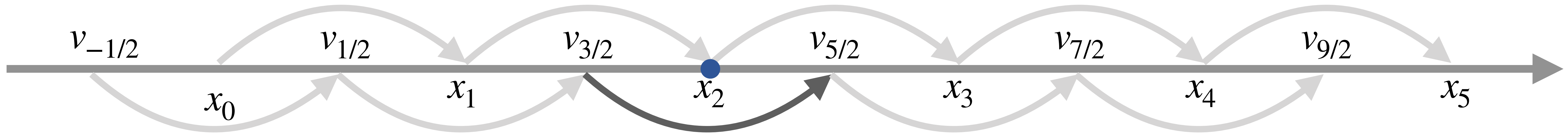
$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0 \,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \frac{a^{2\alpha+1}}{3} \left(\frac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta} \,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

7. Evolution Algorithms: LeapFrog

LeapFrog:

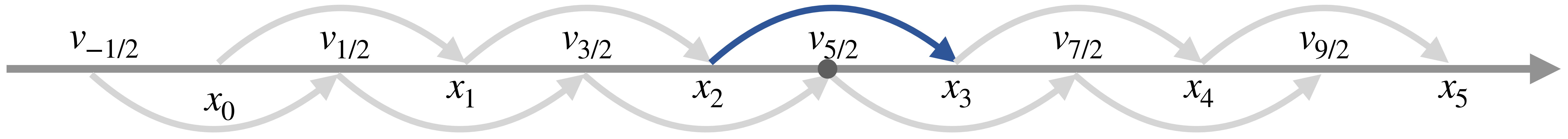


$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta}\,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

LeapFrog:

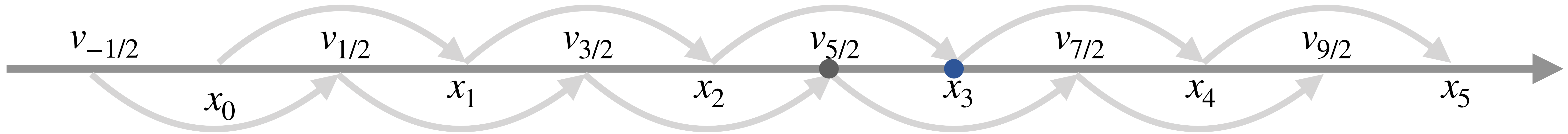


$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha+1)\tilde{E}_V \right] \delta\tilde{\eta}\,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

LeapFrog:

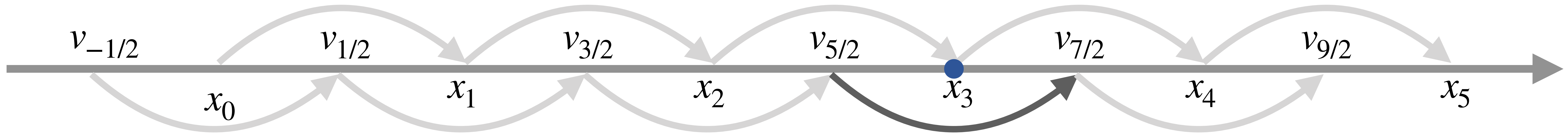


$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\overline{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta}\,, \end{cases} \qquad \overline{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

LeapFrog:



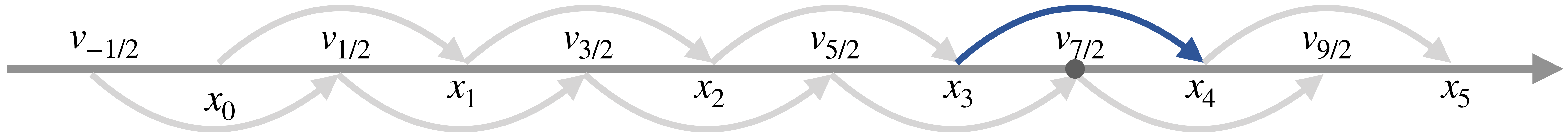
$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta}\,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

7. Evolution Algorithms: LeapFrog

LeapFrog:



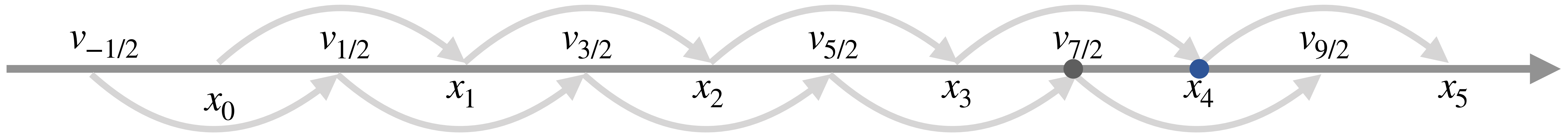
$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha+1)\tilde{E}_V \right] \delta\tilde{\eta}\,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

7. Evolution Algorithms: LeapFrog

LeapFrog:



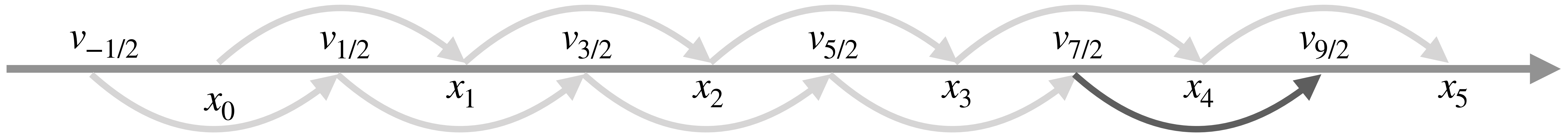
$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0 \,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta} \,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

7. Evolution Algorithms: LeapFrog

LeapFrog:

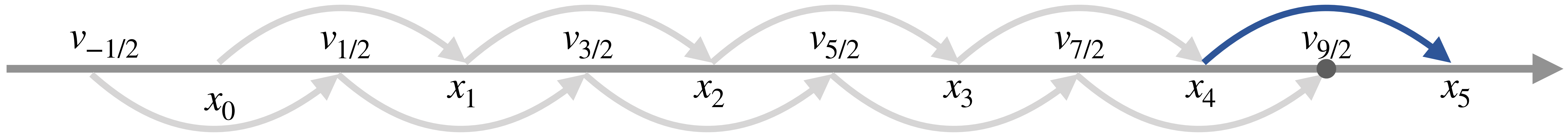


$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta}\,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \qquad \longrightarrow \qquad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

LeapFrog:

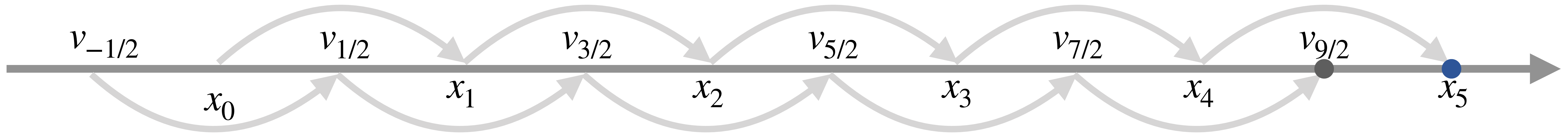


$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0 \,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta} \,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

LeapFrog:



$$IC : \{\tilde{\phi}, a\} \text{ at } \tilde{\eta}_0\,, \quad \{\tilde{\pi}_{-0/2}, b_{-0/2}\} \text{ at } \tilde{\eta}_{-0/2} = \tilde{\eta}_0 - 0.5\delta\tilde{\eta}$$

$$\begin{cases} \tilde{\pi}_{+0/2} = \tilde{\pi}_{-0/2} + \left(a^{1+\alpha} \sum_i \tilde{\nabla}_i^- \tilde{\nabla}_i^+ \tilde{\phi}_{+0} - a^{3+\alpha} \tilde{V}_{,\tilde{\phi}} \right) \delta\tilde{\eta} \\ b_{+0/2} = b_{-0/2} + \dfrac{a^{2\alpha+1}}{3} \left(\dfrac{f_*}{m_p} \right)^2 \left[(\alpha - 2)\bar{\tilde{E}}_K + \alpha \tilde{E}_G + (\alpha + 1)\tilde{E}_V \right] \delta\tilde{\eta}\,, \end{cases} \qquad \bar{\tilde{E}}_K \equiv (\tilde{E}_{K,-0/2} + \tilde{E}_{K,+0/2})$$

$$\begin{cases} a_{+0} = a + b_{+0/2}\delta\tilde{\eta} \quad \longrightarrow \quad a_{+0/2} = (a + a_{+0})/2 \\ \tilde{\phi}_{+0} = \tilde{\phi} + a_{+0/2}^{-(3-\alpha)} \tilde{\pi}_{+0/2} \delta\tilde{\eta} \end{cases}$$

7. Evolution Algorithms

Evolution algorithms:

- Symplectic integrators

  → good for Hamiltonian systems, conserve energy

  examples: LeapFrog, Verlet methods, Yoshida, …

- Non-symplectic integrators

  → broader applicability, e.g. non-Hamiltonian, dissipative systems

  examples: Runge-Kutta methods, …

7. Evolution Algorithms

Evolution algorithms:

- Symplectic integrators

  → good for Hamiltonian systems, conserve energy

  examples: LeapFrog, Verlet methods, Yoshida, …

  → synchronised LeapFrog

- Non-symplectic integrators

  → broader applicability, e.g. non-Hamiltonian, dissipative systems

  examples: Runge-Kutta methods, …

7. Evolution Algorithms

Evolution algorithms:

- Symplectic integrators

  ➔ good for Hamiltonian systems, conserve energy

  examples: LeapFrog, Verlet methods, Yoshida, …

  $\mathcal{O}(dt^2)$ synchronised LeapFrog $\mathcal{O}(dt^n)$

- Non-symplectic integrators

  ➔ broader applicability, e.g. non-Hamiltonian, dissipative systems

  examples: Runge-Kutta methods, …

  ➔ $\mathscr{K}[f(\mathbf{x},\eta),\pi(\mathbf{x},\eta)\}]$

8. Initial Conditions

8. Initial Conditions

How do we set the initial conditions?

8. Initial Conditions

How do we set the initial conditions?

- Arbitrary power spectrum

8. Initial Conditions

How do we set the initial conditions?

- Arbitrary power spectrum
- Spatial configuration

8. Initial Conditions

How do we set the initial conditions?

- Arbitrary power spectrum
- Spatial configuration
- Homogeneous mode + spatially varying fluctuations

8. Initial Conditions

How do we set the initial conditions?

- Arbitrary power spectrum
- Spatial configuration
- Homogeneous mode + spatially varying fluctuations

$$\phi(x,t_*) = \underbrace{\bar{\phi}_*}_{\text{homogeneous mode}} + \underbrace{\delta\phi_*(x)}_{\text{fluctuations}}\,, \qquad \phi'(x,t_*) = \bar{\phi}'_* + \delta\phi'_*(x)$$

8. Initial Conditions

We are interested to impose a set of scalar fluctuations that mimic quantum fluctuations.

8. Initial Conditions

We are interested to impose a set of scalar fluctuations that mimic quantum fluctuations.

- In the continuum, these fluctuations are characterised by:

$$\langle \delta\tilde{\phi}^2 \rangle = \frac{1}{2\pi^2}\int d\log\tilde{k}\,\tilde{k}^3\,\widetilde{\mathcal{P}}_{\delta\tilde{\phi}}(\tilde{k}) \qquad \widetilde{\mathcal{P}}_{\delta\tilde{\phi}}(\tilde{k}) \equiv \frac{1}{2a^2\tilde{\omega}_{\tilde{k},\tilde{\phi}}}\,, \qquad \tilde{\omega}_{\tilde{k},\tilde{\phi}} = \sqrt{\tilde{k}^2 + a^2\tilde{m}^2_{\tilde{\phi}}}\,, \qquad \tilde{m}^2_{\tilde{\phi}} \equiv \left.\frac{\partial^2 \tilde{V}}{\partial\tilde{\phi}^2}\right|_{\tilde{\phi}=\tilde{\phi}_*}$$

(comoving frequency) (effective mass)

8. Initial Conditions

We are interested to impose a set of scalar fluctuations that mimic quantum fluctuations.

- In the continuum, these fluctuations are characterised by:

$$\langle \delta\tilde{\phi}^2 \rangle = \frac{1}{2\pi^2}\int d\log\tilde{k}\,\tilde{k}^3\,\widetilde{\mathcal{P}}_{\delta\tilde{\phi}}(\tilde{k}) \qquad \widetilde{\mathcal{P}}_{\delta\tilde{\phi}}(\tilde{k}) \equiv \frac{1}{2a^2\tilde{\omega}_{\tilde{k},\tilde{\phi}}}\,, \qquad \tilde{\omega}_{\tilde{k},\tilde{\phi}} = \sqrt{\tilde{k}^2 + a^2\tilde{m}^2_{\tilde{\phi}}}\,, \qquad \tilde{m}^2_{\tilde{\phi}} \equiv \left.\frac{\partial^2\tilde{V}}{\partial\tilde{\phi}^2}\right|_{\tilde{\phi}=\tilde{\phi}_*}$$

(comoving frequency) (effective mass)

- On the lattice, such fluctuations are then characterised by:

$$\langle f^2\rangle_V = \frac{1}{2\pi^2}\sum_{|\tilde{\mathbf{n}}|}\Delta\log k(\tilde{\mathbf{n}})k^3(\tilde{\mathbf{n}})\frac{L^3}{N^6}\langle\,|f(\tilde{\mathbf{n}})|^2\,\rangle_{R(\tilde{\mathbf{n}})}\,, \qquad \left\langle\,|f(\tilde{\mathbf{n}})|^2\,\right\rangle_{R(\tilde{\mathbf{n}})} = \left(\frac{\omega_*}{f_*}\right)^2\left(\frac{N}{\delta\tilde{x}}\right)^2\widetilde{\mathcal{P}}_{\delta\tilde{\phi}}(\tilde{k})$$

8. Initial Conditions

At each lattice point in **momentum space** we impose the following sum of left- and right-moving waves:

$$\delta\tilde{\phi}(\tilde{\mathbf{n}}) = \frac{1}{\sqrt{2}}\left(|\delta\tilde{\phi}_1(\tilde{\mathbf{n}})|e^{i\alpha_1(\tilde{\mathbf{n}})} + |\delta\tilde{\phi}_2(\tilde{\mathbf{n}})|e^{i\alpha_2(\tilde{\mathbf{n}})}\right)$$

$$\delta\tilde{\phi}'(\tilde{\mathbf{n}}) = \frac{i\tilde{\omega}_{k,\phi}}{\sqrt{2}a^{1-\alpha}}\left(|\delta\tilde{\phi}_1(\tilde{\mathbf{n}})|e^{i\alpha_1(\tilde{\mathbf{n}})} + |\delta\tilde{\phi}_2(\tilde{\mathbf{n}})|e^{i\alpha_2(\tilde{\mathbf{n}})}\right) - \tilde{\mathcal{H}}\delta\tilde{\phi}(\tilde{\mathbf{n}})$$

At each lattice point in **momentum space** we impose the following sum of left- and right-moving waves:

$$\delta\tilde{\phi}(\tilde{\mathbf{n}}) = \frac{1}{\sqrt{2}}(|\delta\tilde{\phi}_1(\tilde{\mathbf{n}})|e^{i\alpha_1(\tilde{\mathbf{n}})} + |\delta\tilde{\phi}_2(\tilde{\mathbf{n}})|e^{i\alpha_2(\tilde{\mathbf{n}})})$$

$$\delta\tilde{\phi}'(\tilde{\mathbf{n}}) = \frac{i\tilde{\omega}_{k,\phi}}{\sqrt{2}a^{1-\alpha}}(|\delta\tilde{\phi}_1(\tilde{\mathbf{n}})|e^{i\alpha_1(\tilde{\mathbf{n}})} + |\delta\tilde{\phi}_2(\tilde{\mathbf{n}})|e^{i\alpha_2(\tilde{\mathbf{n}})}) - \tilde{\mathscr{H}}\delta\tilde{\phi}(\tilde{\mathbf{n}})$$

- $\alpha_1(\tilde{\mathbf{n}}), \alpha_2(\tilde{\mathbf{n}})$: Random phases in the range $[0,2\pi)$
- $\delta\tilde{\phi}_1(\tilde{\mathbf{n}}), \delta\tilde{\phi}_2(\tilde{\mathbf{n}})$: Rayleigh distributions with expected square amplitude given by:

$$\left\langle |f(\tilde{\mathbf{n}})|^2 \right\rangle_{R(\tilde{\mathbf{n}})} = \left(\frac{\omega_*}{f_*}\right)^2 \left(\frac{N}{\delta\tilde{x}}\right)^2 \frac{1}{2a^2\sqrt{\tilde{k}^2(\tilde{\mathbf{n}}) + a^2\tilde{m}_{\tilde{\phi}}^2}},$$

At each lattice point in **momentum space** we impose the following sum of left- and right-moving waves:

$$\delta\tilde{\phi}(\tilde{\mathbf{n}}) = \frac{1}{\sqrt{2}}(|\delta\tilde{\phi}_1(\tilde{\mathbf{n}})|e^{i\alpha_1(\tilde{\mathbf{n}})} + |\delta\tilde{\phi}_2(\tilde{\mathbf{n}})|e^{i\alpha_2(\tilde{\mathbf{n}})})$$

$$\delta\tilde{\phi}'(\tilde{\mathbf{n}}) = \frac{i\tilde{\omega}_{k,\phi}}{\sqrt{2}a^{1-\alpha}}(|\delta\tilde{\phi}_1(\tilde{\mathbf{n}})|e^{i\alpha_1(\tilde{\mathbf{n}})} + |\delta\tilde{\phi}_2(\tilde{\mathbf{n}})|e^{i\alpha_2(\tilde{\mathbf{n}})}) - \tilde{\mathcal{H}}\delta\tilde{\phi}(\tilde{\mathbf{n}})$$

- $\alpha_1(\tilde{\mathbf{n}}), \alpha_2(\tilde{\mathbf{n}})$: Random phases in the range $[0, 2\pi)$
- $\delta\tilde{\phi}_1(\tilde{\mathbf{n}}), \delta\tilde{\phi}_2(\tilde{\mathbf{n}})$: Rayleigh distributions with expected square amplitude given by:

$$\left\langle |f(\tilde{\mathbf{n}})|^2 \right\rangle_{R(\tilde{\mathbf{n}})} = \left(\frac{\omega_*}{f_*}\right)^2 \left(\frac{N}{\delta\tilde{x}}\right)^2 \frac{1}{2a^2\sqrt{\tilde{k}^2(\tilde{\mathbf{n}}) + a^2\tilde{m}_{\tilde{\phi}}^2}},$$

Gaussian Fluctuations

Questions?