*ab*TEM: transmission electron microscopy from first principles

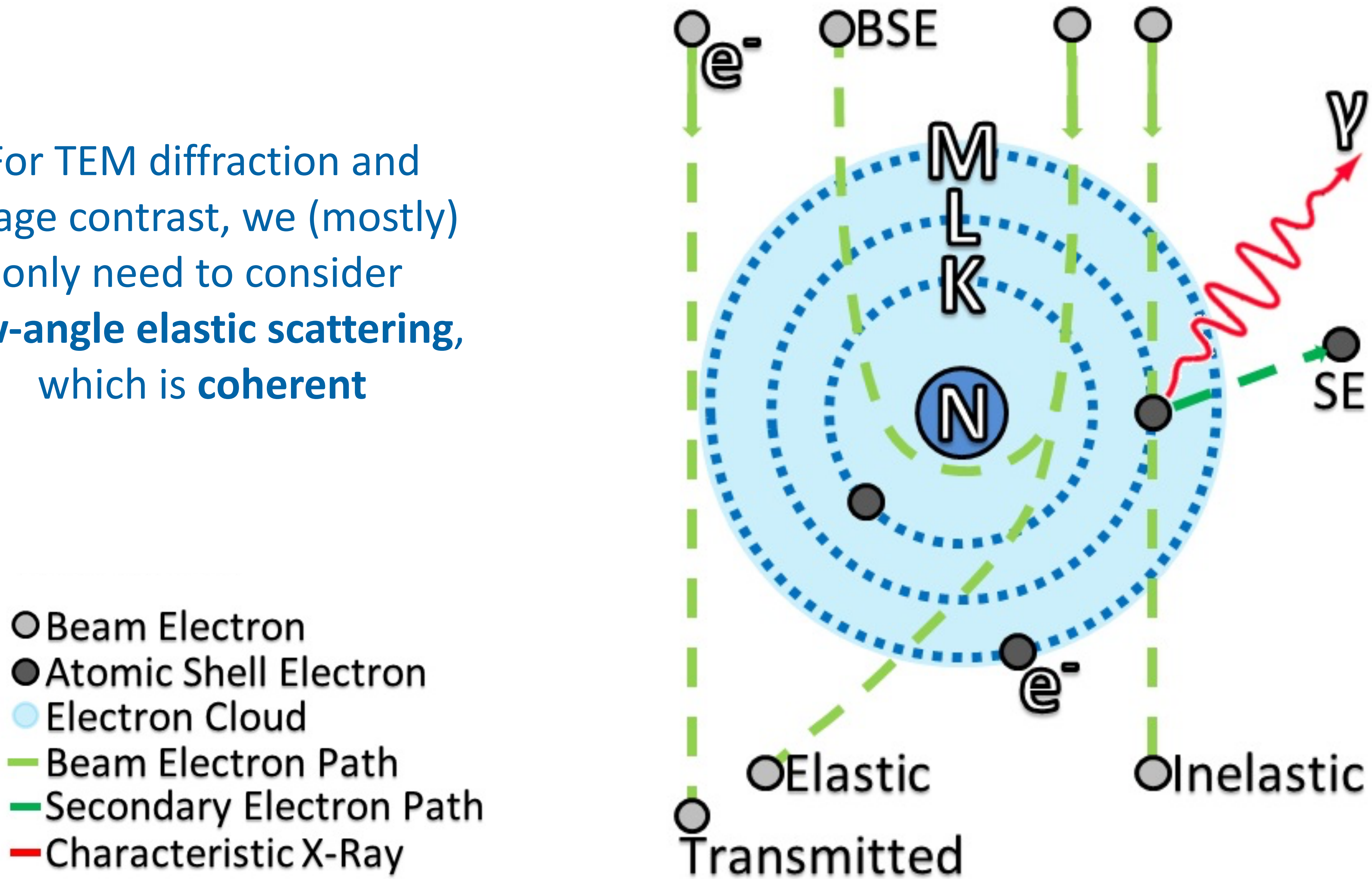
Hyperspy 2025
06 June 2025

Usage poll:

Toma Susi (University of Vienna, Austria)
Jacob Madsen

Electron scattering

For TEM diffraction and image contrast, we (mostly) only need to consider **low-angle elastic scattering**, which is **coherent**

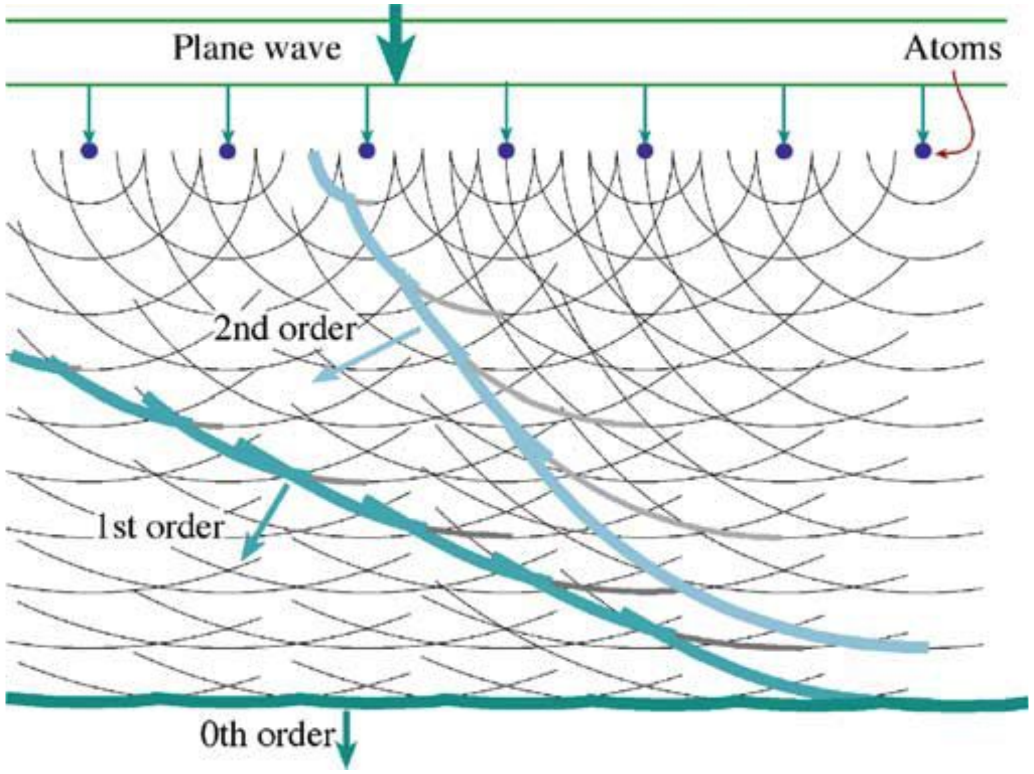


https://en.wikipedia.org/wiki/Electron_scattering

Theory of elastic scattering

Quantum mechanical description

- Angular distribution of scattered intensity
  - For atoms, *atomic form factor, f*
  - For crystals, *structure factor, F*
    - Preferred directions due to constructive interference



- Full analytical description in general not feasible
  - Quantum mechanical many-body problem
  - Approximate analytical solution can be obtained as a *Fourier transform of the potential*
  - In general: numerical simulation of scattering

Williams & Carter

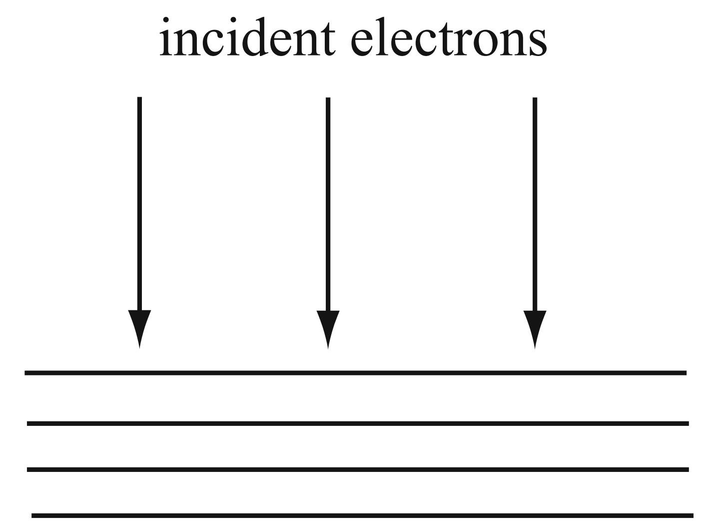
Scattering simulations

- Schrödinger equation for electron wave transmitting through specimen
  - Including time evolution explicitly would be prohibitively expensive
    - Very small wavelength/high frequency = very small sampling
  - We can use time-independent Schrödinger equation instead
- Some approximation are still needed
  - “Fast electrons” = initial kinetic energy >> energy change in specimen
    - Motion predominantly in the *z* direction with small perturbation
  - Non-relativistic equations
    - However, with correct relativistic mass and wavelength
    - Fine at ~100 keV, less so at 1 MeV (latter nowadays quite rare)
- Numerical implementation: fast Fourier transforms!

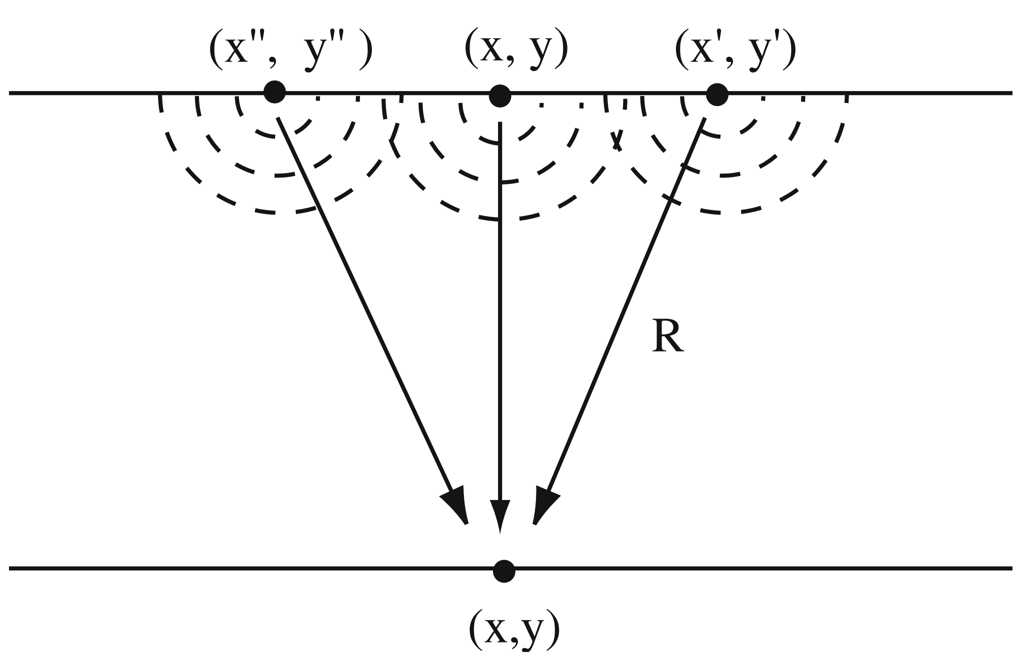
Numerical implementations

Scattering simulations

- Multiple formalisms available
  - FFT
  - Bloch-wave *(added to abTEM!)*
  - real-space *(standard in abTEM)*
  - PRISM *(fastest in abTEM :)*
- Most commonly: real-space simulations based on **multislice algorithm**
  - Sample potential divided into thin slices
  - Project, transmit, propagate…
    - Modify *transmission* step to include magnetic vector potential



c) transmit and propagate



Williams & Carter

Weak phase object

Wavefunction for plane wave moving along *z* axis

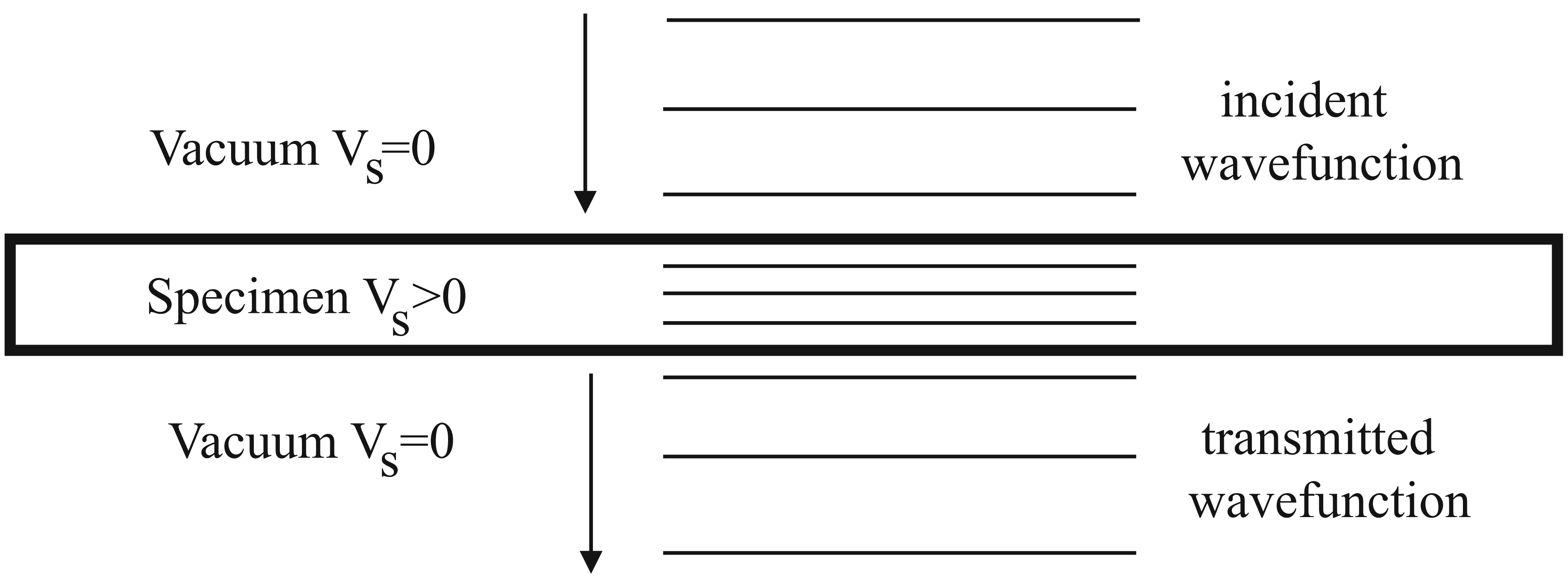
$$\psi(\mathbf{x}) = \exp(2\pi i k_z z) = \exp(2\pi i z/\lambda)$$

wavevector

$$k_z = \frac{1}{\lambda} = \frac{\sqrt{eV(2m_0c^2 + eV)}}{hc}$$

eV = electron kinetic energy

For *eV* >> V_s potential energy in the specimen,
its effect is to cause a small change in wavelength



Phase shift

Wavelength in specimen

$$\frac{1}{\lambda_s} = \frac{[(eV + eV_s)(2m_0c^2 + eV + eV_s)]^{1/2}}{hc}$$

$$= \frac{[eV(2m_0c^2 + eV) + eV_s(2m_0c^2 + 2eV + eV_s)]^{1/2}}{hc}$$

$$= \frac{1}{\lambda}\left[1 + \frac{eV_s(2m_0c^2 + 2eV + eV_s)}{eV(2m_0c^2 + eV)}\right]^{1/2}$$

Expanding and keeping leading order in V_s/V

$$\frac{1}{\lambda_s} \sim \frac{1}{\lambda}\left[1 + \frac{eV_s(2m_0c^2 + 2eV)}{2eV(2m_0c^2 + eV)} + \cdots\right] \sim k_z + \frac{V_s(m_0c^2 + eV)}{\lambda V(2m_0c^2 + eV)} + \cdots$$

Changing wavelength equivalent to a phase shift

$$\psi(\mathbf{x}) \sim \exp(2\pi i k_z z)\exp(i\sigma_e V_s z)$$

interaction parameter

$$\sigma_e = \frac{2\pi}{\lambda V}\left(\frac{m_0c^2 + eV}{2m_0c^2 + eV}\right) = \frac{2\pi m e\lambda}{h^2}$$

relativistic mass $m = \gamma m_0$

Transmission through thin slices

For very thin specimen, phase shift = integral of potential

$$\psi_t(\mathbf{x}) = t(\mathbf{x})\exp(2\pi i k_z z)$$
$$t(\mathbf{x}) = \exp[i\sigma_e v_z(\mathbf{x})]$$

t = transmission function

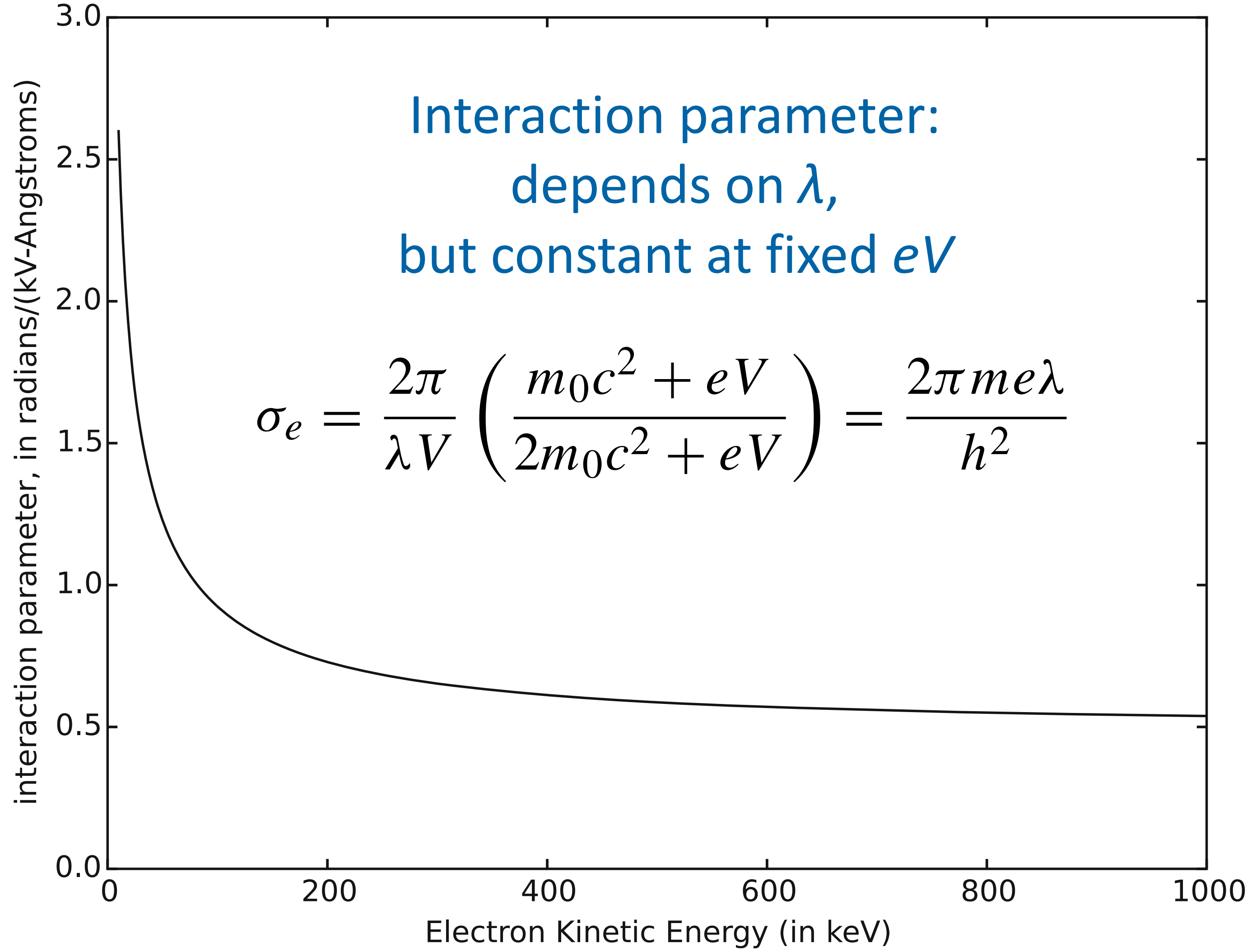
with the projected potential

$$v_z(\mathbf{x}) = v_z(x, y) = \int V_s(x, y, z)dz$$

Weak phase object approx. (WPOA):

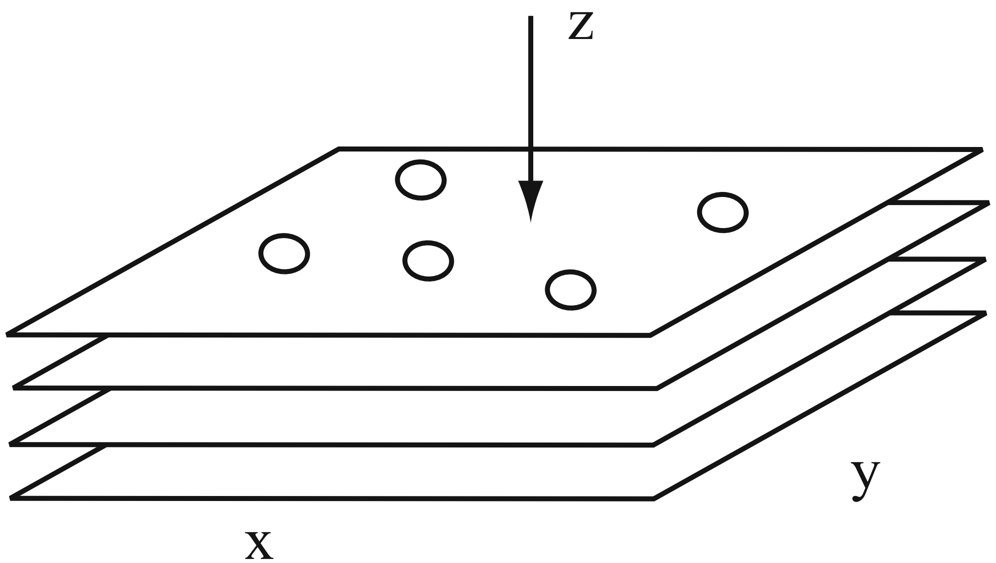
1) $V_S << eV$
2) 3D potential ⇒ 2D projection

Does NOT describe most specimens…
… but DOES describe potential slices!



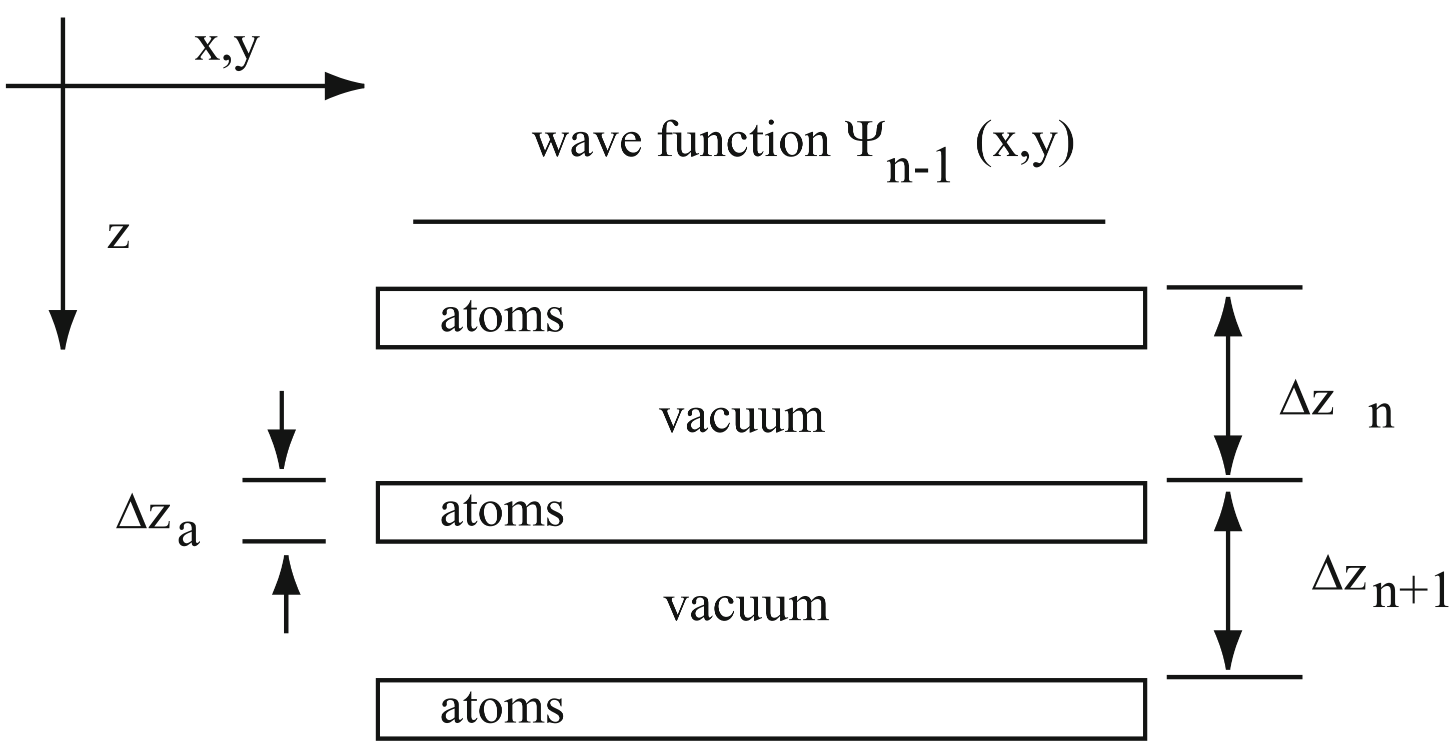
$$\sigma_e = \frac{2\pi}{\lambda V}\left(\frac{m_0c^2 + eV}{2m_0c^2 + eV}\right) = \frac{2\pi m e\lambda}{h^2}$$

Slicing the specimen



Each slice = thin enough for WPOA to hold

Within each slice: rectangular cell with periodic boundary conditions



Ideally: align atomic layers with the slice boundaries

If there is periodic structure along *z*, pre-calculating and storing transmission function is computationally efficient

Discrete numerical simulations

- An image needs to be represented as numerical data
  - Two- dimensional array of machine-represented numbers
    - Each number is one pixel or spot in the image
    - Intensity is proportional to numerical value
  - Sufficiently large number of pixels able to represent intensities required
    - Sampling the image has rules and limitations
    - Both spatial sampling and bit depth matter
- Converting between reciprocal and real space theoretically important
  - Fast Fourier transform (FFT): one of the most efficient computer algorithms
    - Computes the Fourier transform of discretely sampled data
  - Image simulation usually organized around FFT to save time –
    choice of simulation method is driven by computational efficiency!

Numerical discretization

Real-space discretization

$$x = i\Delta x \quad i = 0, 1, 2, \cdots, (N_x - 1)$$

$$y = j\Delta y \quad j = 0, 1, 2, \cdots, (N_y - 1)$$

Reciprocal-space discretization

$$k_x = i\Delta k_x \quad i = 0, 1, 2, \ldots, (N_x - 1)$$

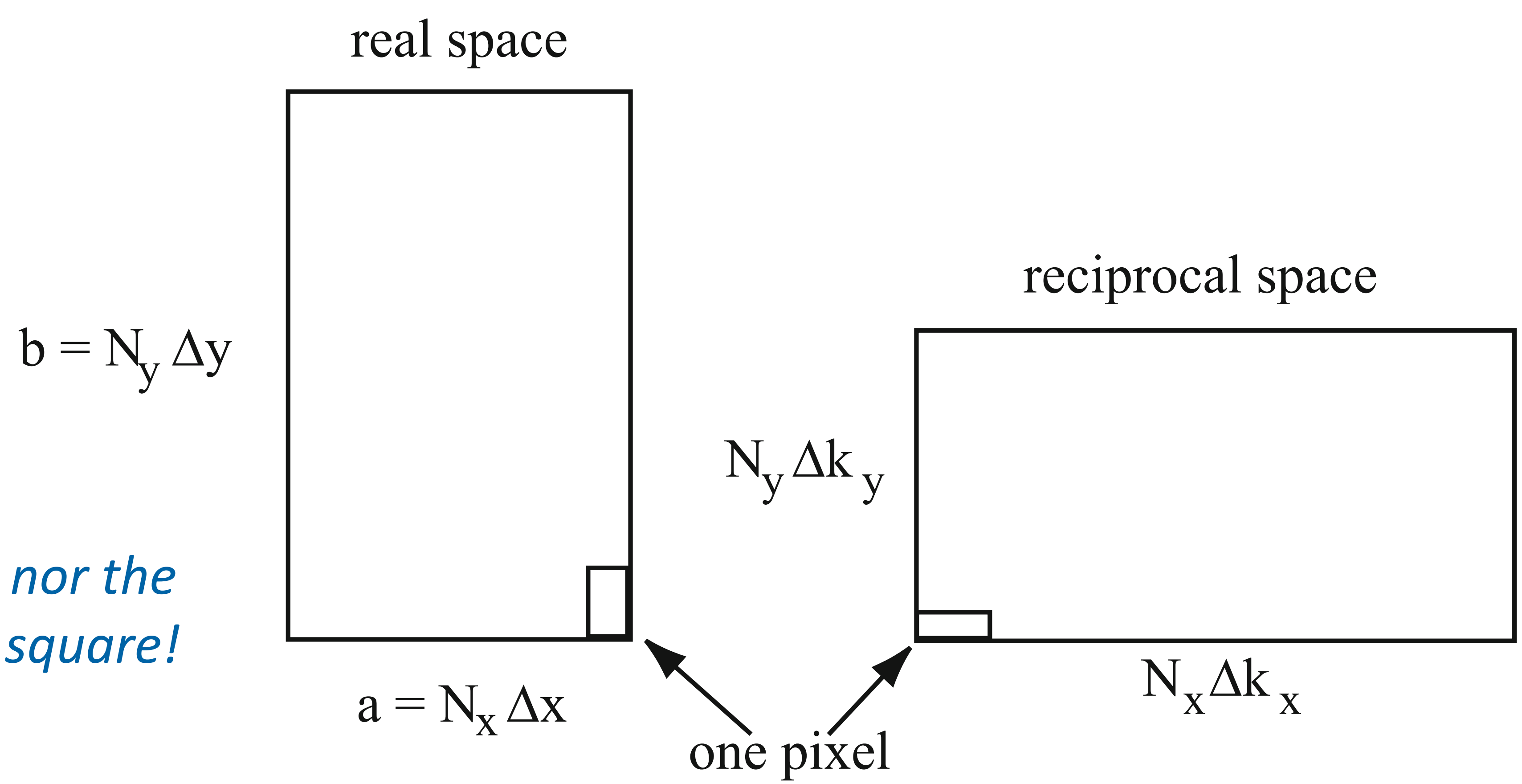
$$k_y = j\Delta k_y \quad j = 0, 1, 2, \ldots, (N_y - 1)$$

Real-space sampling

$\Delta x = a/N_x$ and $\Delta y = b/N_y$

Reciprocal-space sampling

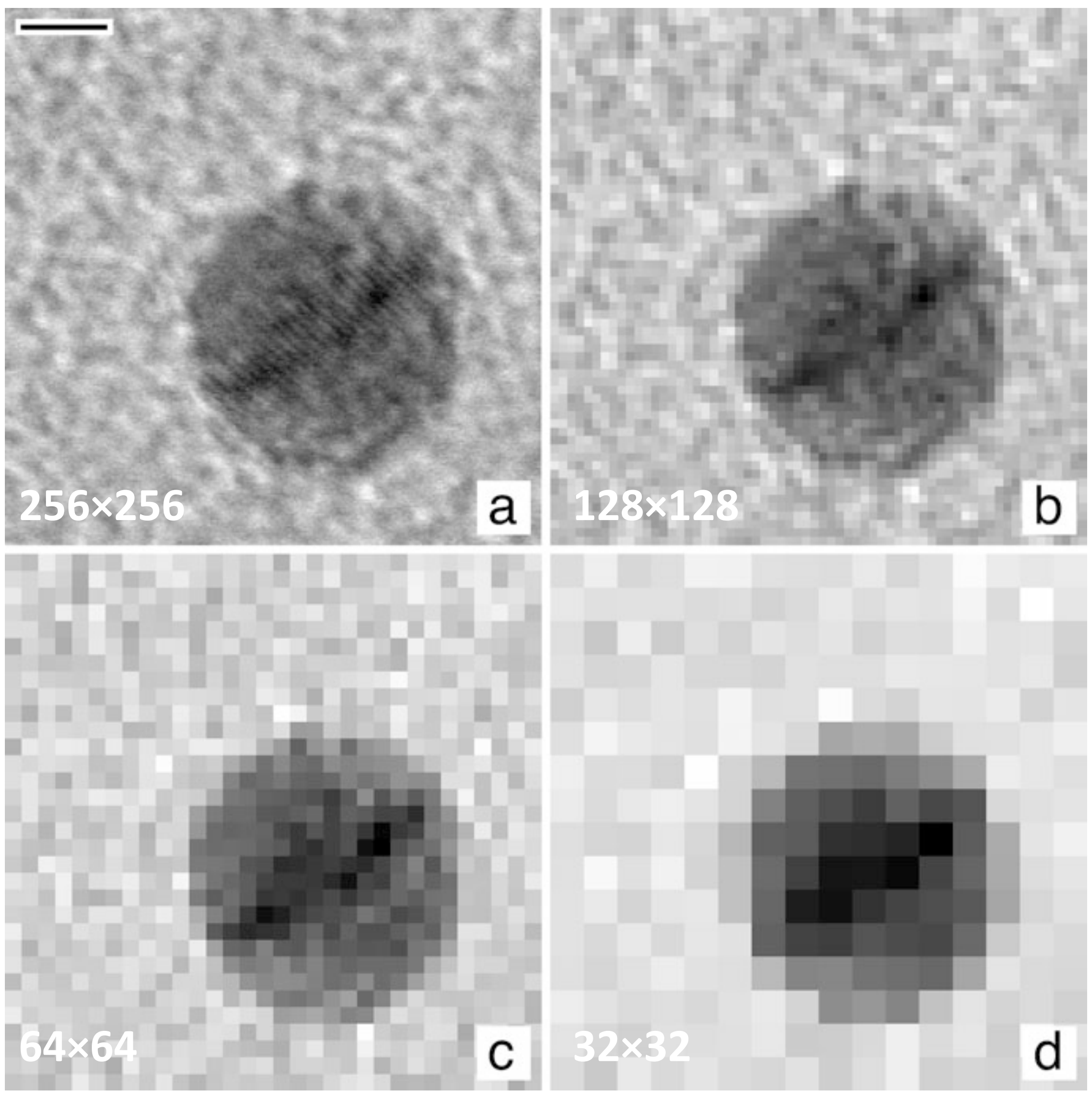
$\Delta k_x = 1/a$ and $\Delta k_y = 1/b$



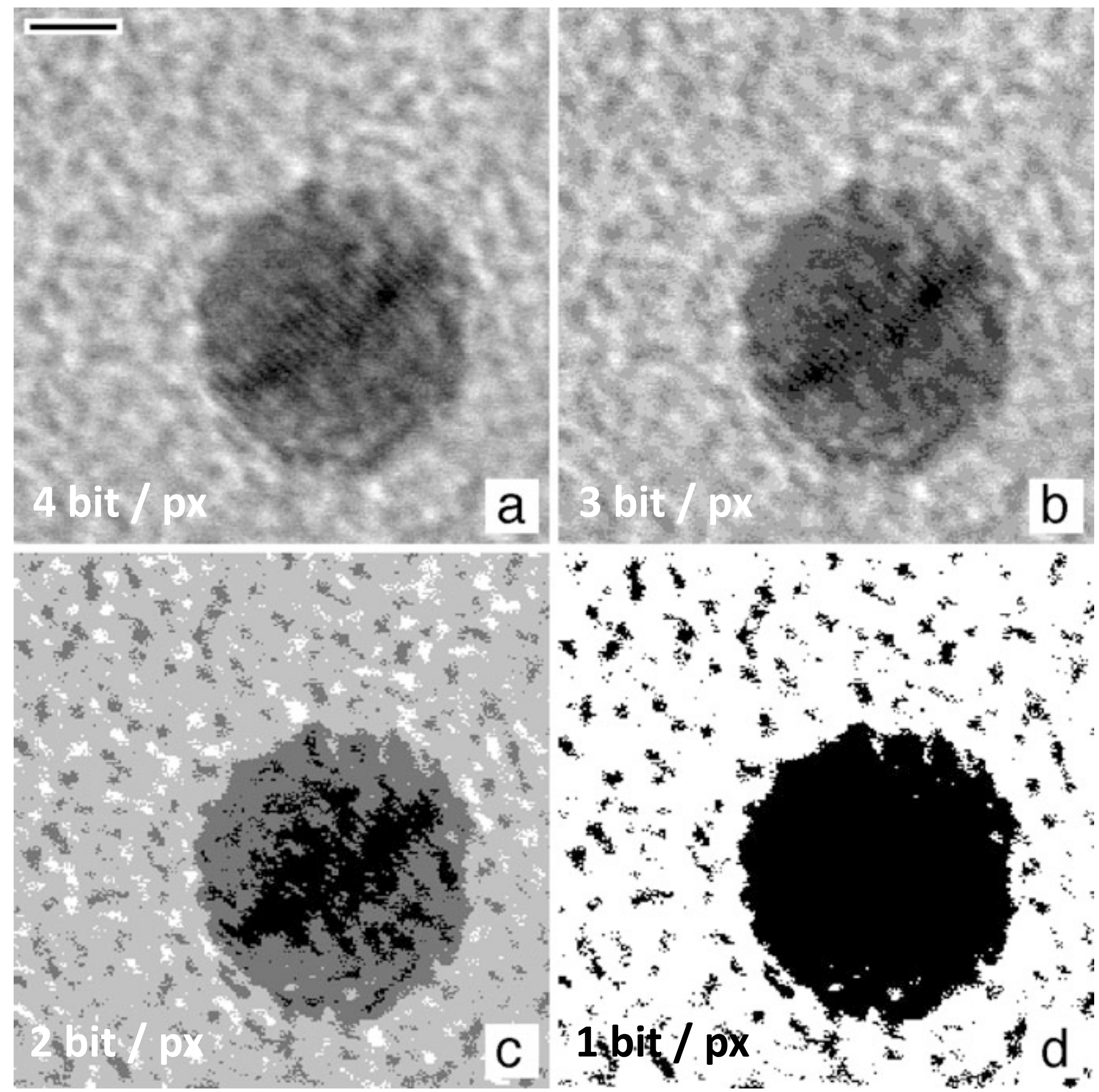
Neither the pixels nor the image have to be square!

Pixel sampling and bit depth

Experimental: as many pixels as in camera/scan



Typically store 32-bit, display: 8-bit



<u>example</u>: STEM image of gold particle on amorphous carbon

Sampling

Maximum spatial frequency

$$|k_x| < \frac{1}{2\Delta x}$$

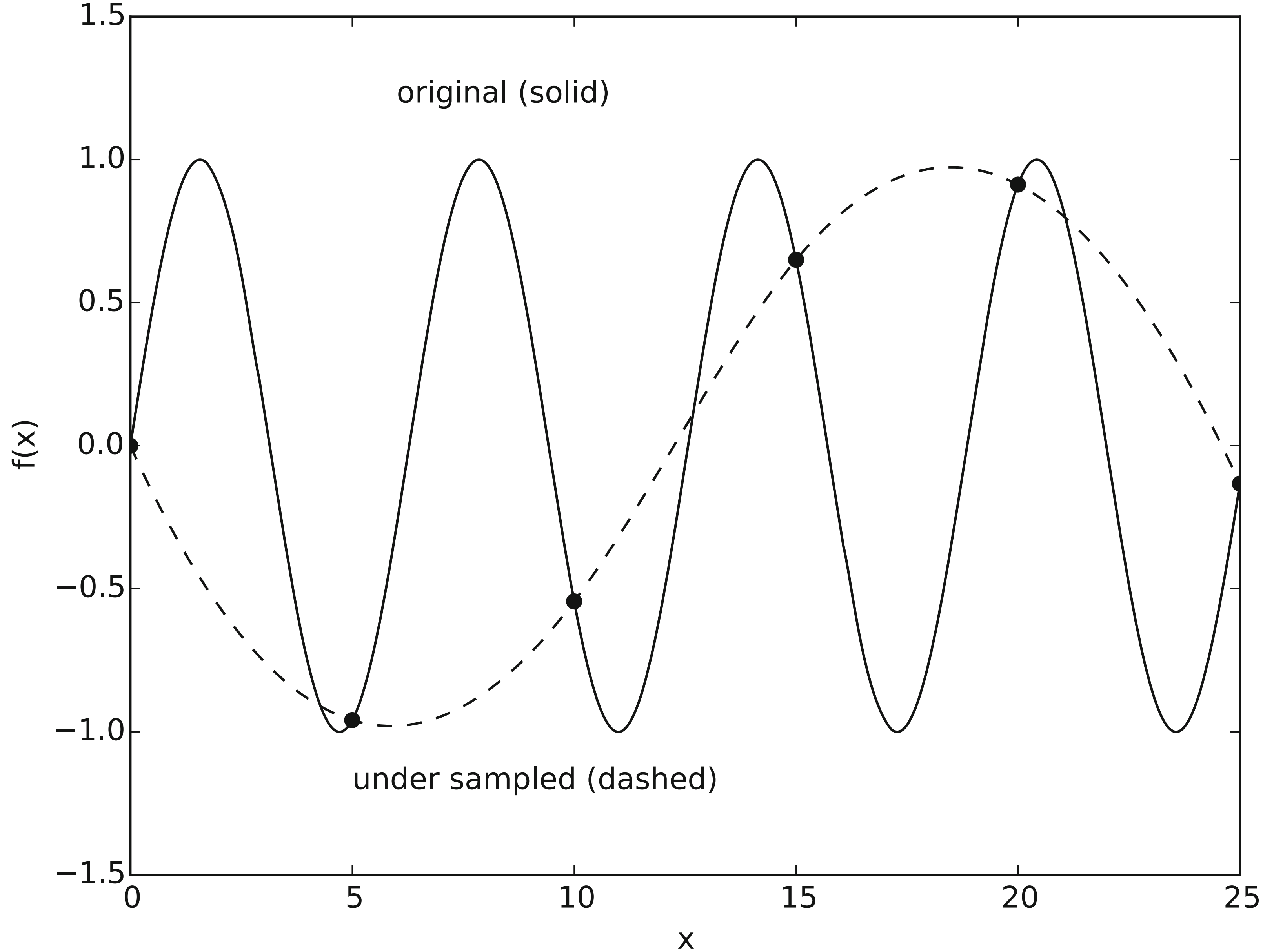
$$|k_y| < \frac{1}{2\Delta y}$$

This is the **Nyquist limit**.

Sampling less densely will lead to loss of information.

Sampling more densely may be redundant (abTEM can in many cases interpolate).

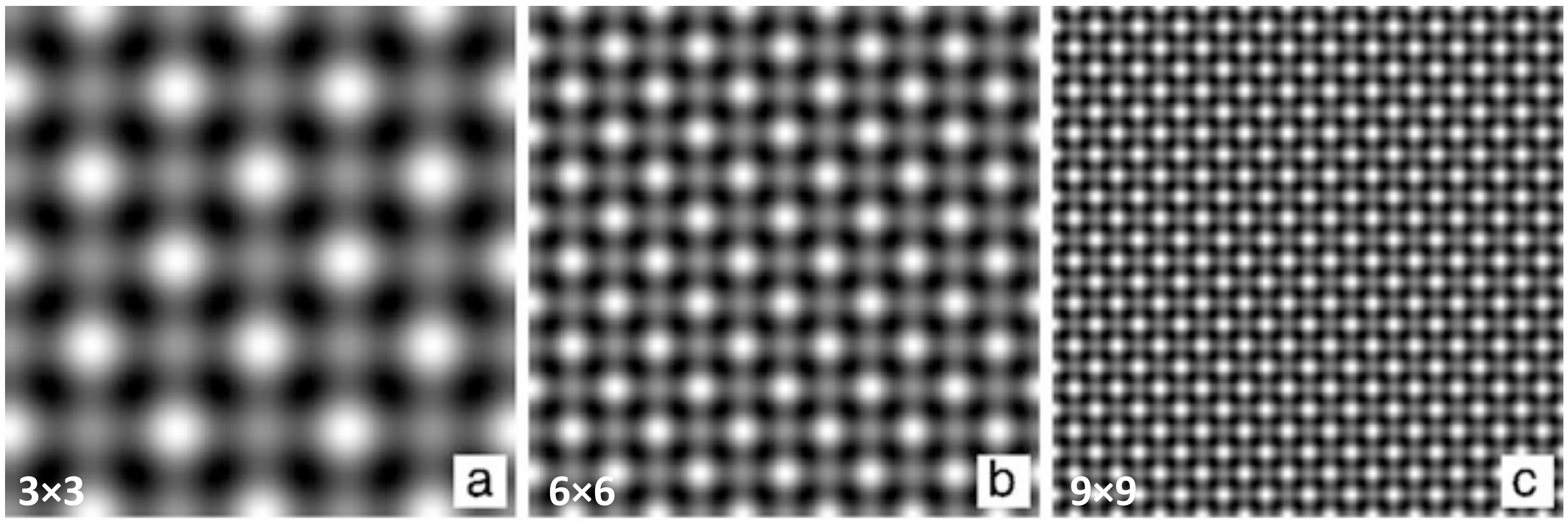
Effect of under-sampling



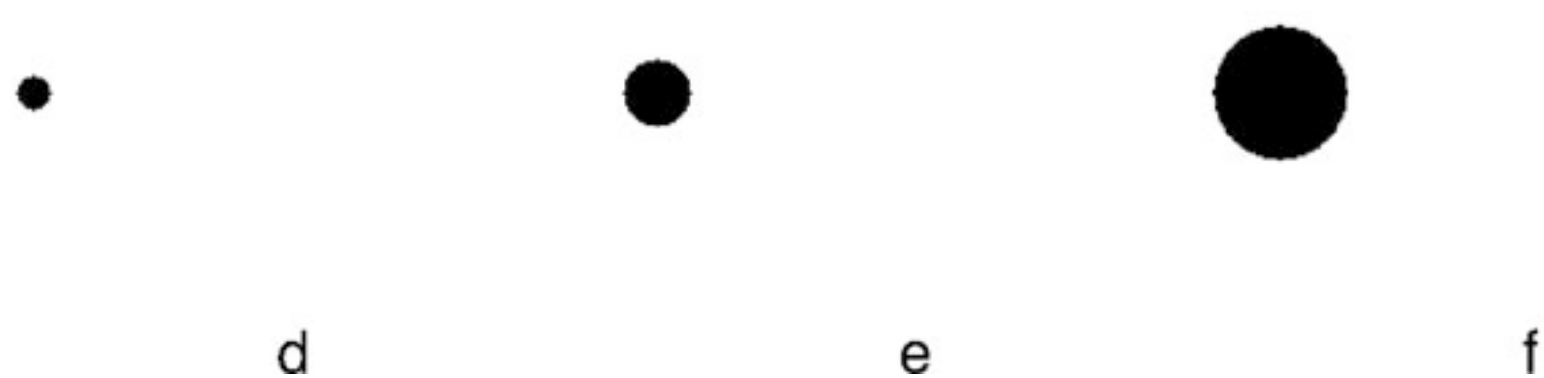
high frequencies get “aliased” as low frequencies

Real vs. reciprocal space sampling

Increasing real-space supercell size at a fixed 256×256 pixels



Sampling in real space and reciprocal space are **inversely coupled**.



Objective aperture sampling in reciprocal space increases (more pixels)

<u>example</u>: TEM image simulations of Sr_3TiO_2 supercells

Independent atom models

universität wien

Calculate and parametrize

- Calculate atomic all-electron density using first-principles modeling (e.g. Hartree-Dirac-Fock, all-electron DFT, etc)
- Choose functional form for parametrization

$$f_e(q) = \sum_{i=1}^{N_L} \frac{a_i}{q^2 + b_i} + \sum_{i=1}^{N_G} c_i \exp(-d_i q^2)$$

Kirkland parameterization

- Calculate potential, Fourier-transform ⇒ atomic form factor
- Optimize parameters for each element to fit form factor

$$\chi^2 = \frac{1}{2N_q - 2N_L - 2N_G} \sum_{j=1}^{N_q} \left\{ \left[\frac{f_{xj} - f_x(q_j)}{\sigma_{xj}} \right]^2 + \left[\frac{f_{ej} - f_e(q_j)}{\sigma_{ej}} \right]^2 \right\}$$

minimize fit error

- End result: numerically fast form factors (potentials) for each element

Kirkland

ab initio description of bonding for transmission electron microscopy

Jacob Madsen [a], Timothy J. Pennycook [b], Toma Susi [a,*]

[a] Faculty of Physics, University of Vienna, Boltzmanngasse 5, 1090 Vienna, Austria
[b] EMAT, University of Antwerp, Groenenborgerlaan 171, G.U.431, 2020 Antwerp, Belgium

TS *et al.*, Ultramicroscopy, 197 (2019) 16–22

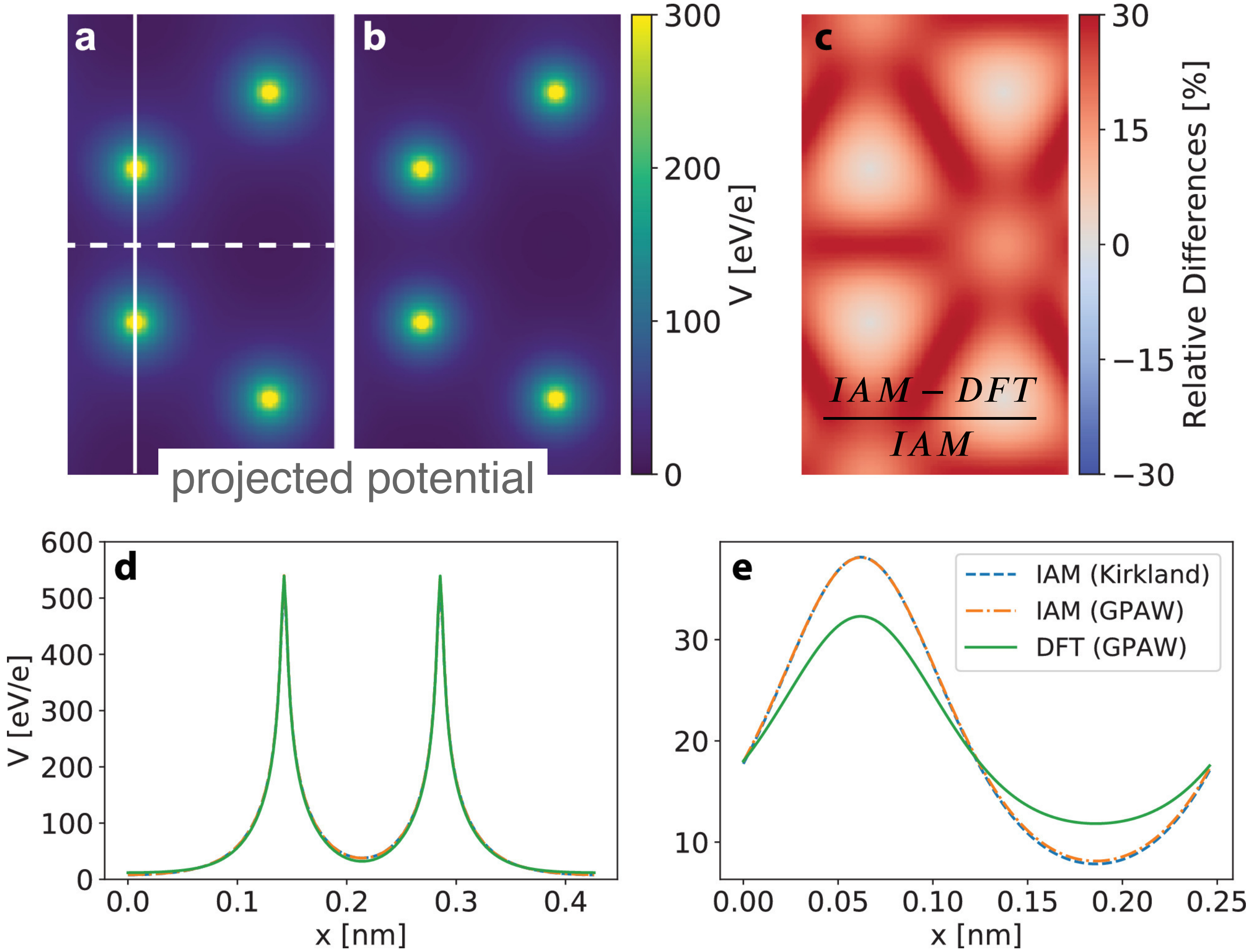


Table 2
ab initio simulations of the electrostatic potential of materials for explicit TEM electron scattering simulations. See the caption of Table 1 for a description of the different treatments of core electrons.

DFT code	Core electrons	Material	Refs.
WIEN2K	FLAPW	MgO	[64]
WIEN2K	FLAPW	graphene, hBN	[6,65]
SIESTA	pseudo	graphene	[66]
QUANTUM ESPRESSO	pseudo	AlN	[67]
EXCITING	FLAPW	graphene	[68]
ABINIT/ELK	pseudo	WSe_2	[69,70]
WIEN2K	FLAPW	GaN	[22]
VASP	pseudo	$SrTiO_3$	[71]
GPAW	PAW	graphene, hBN	[52,72]
CASTEP/WIEN2K	pseudo, FLAPW	hBN	[73]
FPLO-18	FLAPW	graphene, hBN	[74]
GPAW	PAW	GaP	[75]

Many different DFT codes,
many multislice codes…

…but none with direct integration
with *ab initio* scattering potentials

Bonding effects do matter for HRTEM,
holography quantitative ED, 4D-STEM…

ASE + GPAW! + abTEM

J. Madsen, T.J. Pennycook, **TS**, Ultramicroscopy (2021) 113253; 10.1016/j.ultramic.2021.113253

*ab*TEM code: TEM from first principles

Our design goals

1. Integration with atomistic codes

ASE GPAW!

2. Python, but *fast*

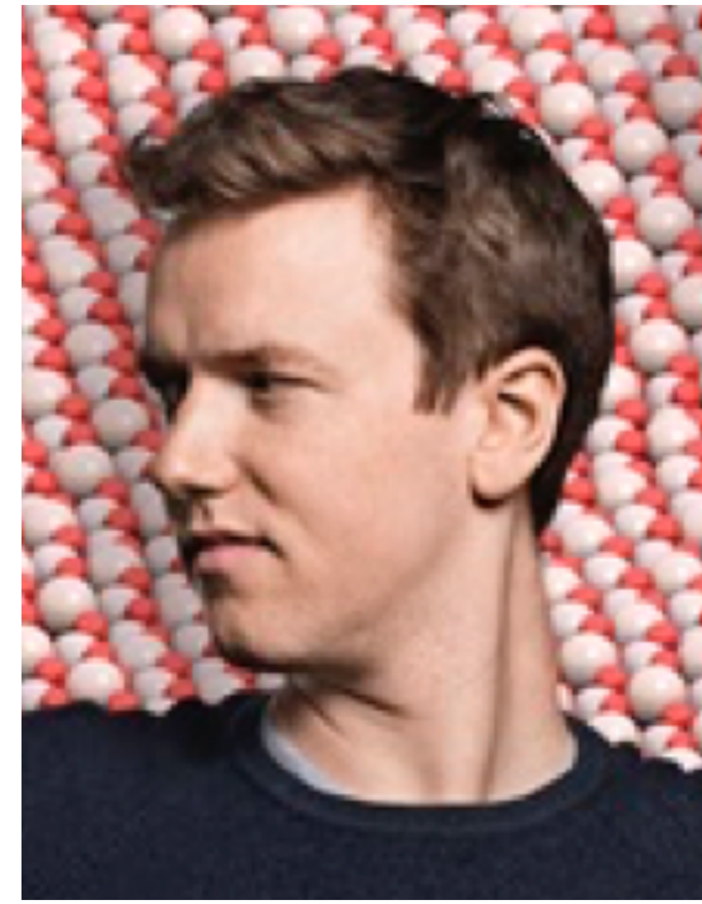
CuPy Numba

3. Promote notebook/scripted workflow

jupyter

4. Access to low-level API: Easy to extend

5. Open source (huge ecosystem!)



Dr. Jacob Madsen

high-performance parallel GPUs & file access

DASK NVIDIA CUDA Zarr

abTEM

github.com/abTEM/abTEM

*ab*TEM's API: interchangeable objects

Atoms

GPAW

Potential

GPAWPotential

PotentialArray

Waves

PlaneWave

Probe

SMatrix

AnnularDetector

SegmentedDetector

PixelatedDetector

WavefunctionDetector

GridScan

LineScan

PositionScan

FrozenPhonons

MDFrozenPhonons

*ab*TEM's API: STEM/ADF image with IAM

Atoms

GPAW

Potential

GPAWPotential

PotentialArray

Waves

PlaneWave

Probe

SMatrix

AnnularDetector

SegmentedDetector

PixelatedDetector

WavefunctionDetector

GridScan

LineScan

PositionScan

FrozenPhonons

MDFrozenPhonons

*ab*TEM's API: 4D-STEM with IAM

Atoms
GPAW

Potential
GPAWPotential
PotentialArray

Waves
PlaneWave
Probe
SMatrix

AnnularDetector
SegmentedDetector
PixelatedDetector
WavefunctionDetector

GridScan
LineScan
PositionScan

FrozenPhonons
MDFrozenPhonons

https://abtem.readthedocs.io/en/latest/user_guide/tutorials/charge_density.html

*ab*TEM's API: 4D-STEM with DFT

Atoms

GPAW

Potential

GPAWPotential

PotentialArray

Waves

PlaneWave

Probe

SMatrix

AnnularDetector

SegmentedDetector

PixelatedDetector

WavefunctionDetector

GridScan

LineScan

PositionScan

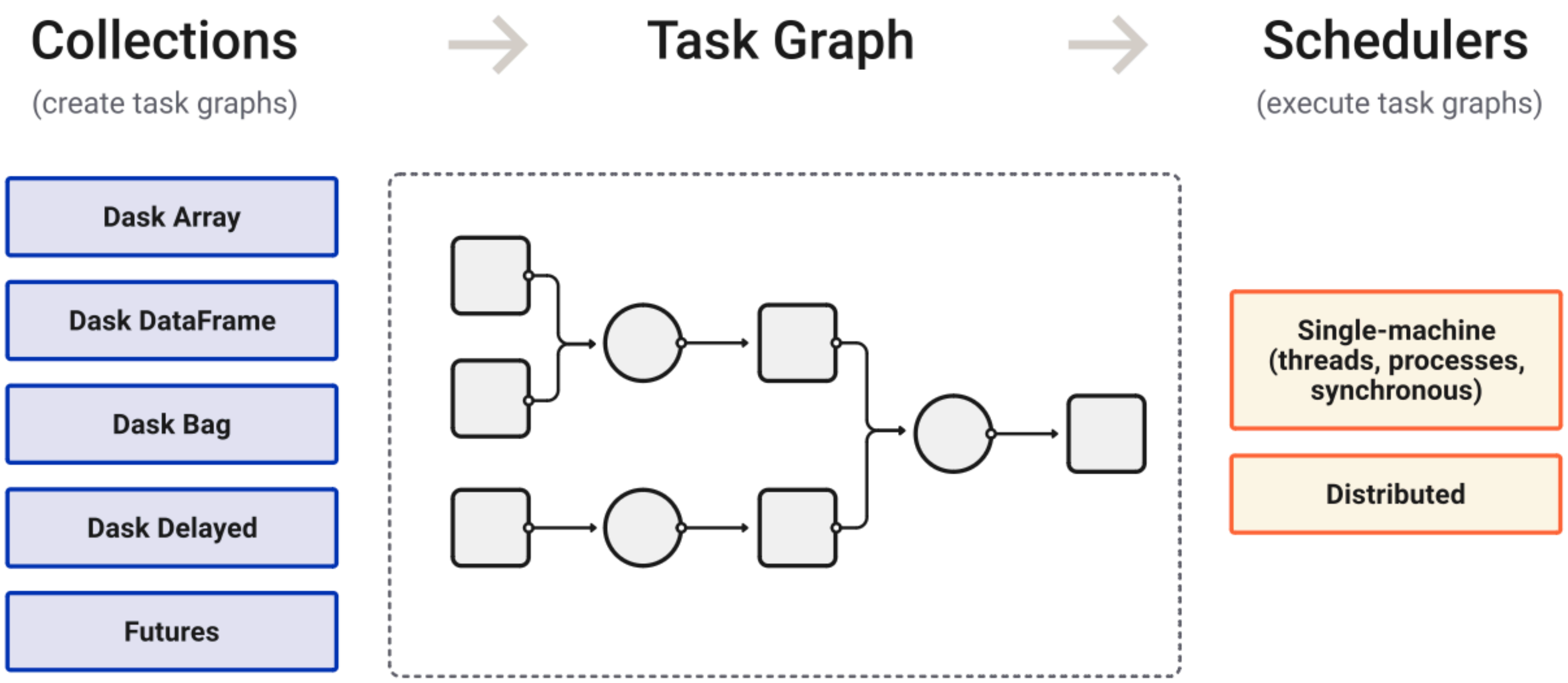
FrozenPhonons

MDFrozenPhonons

https://abtem.readthedocs.io/en/latest/user_guide/walkthrough/scan_and_detect.html abTEM with DASK

Task graphs

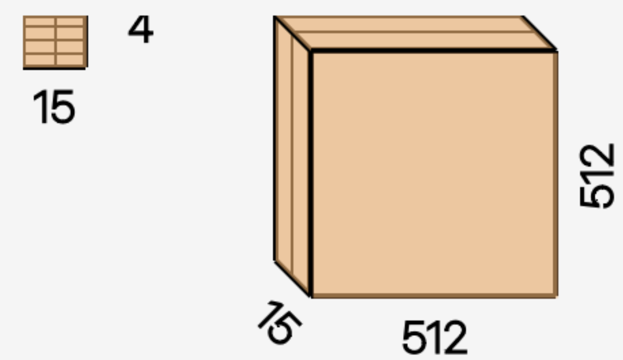
Simulating TEM experiments requires executing multiple *tasks* where each one may depend on the output of previous tasks. In Dask this is represented as a *task graph*, where each task is a *node*, with *edges* between nodes if it is dependent on another task. The simulation result is obtained by executing each task (node) in the graph with a Dask *scheduler* on a single machine or a cluster.

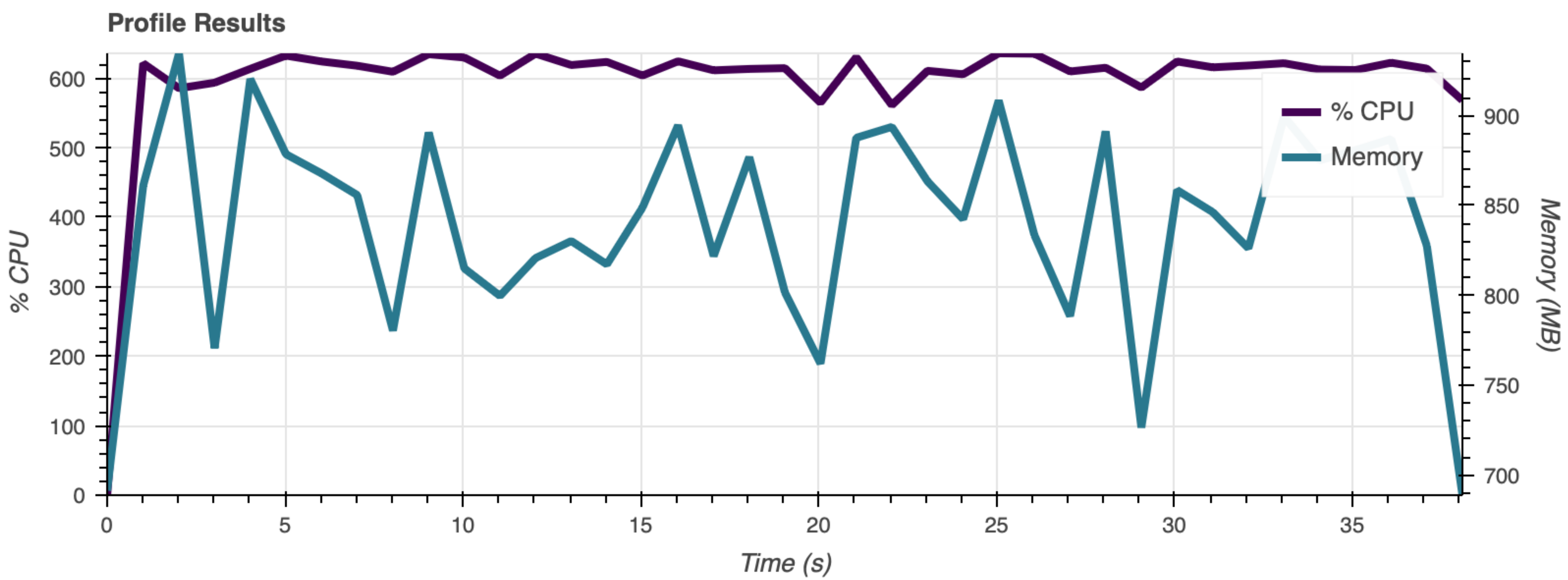


https://abtem.github.io/doc/user_guide/walkthrough/parallelization.html

```
type                label             coordinates
-----------------   --------------    --------------------
FrozenPhononsAxis   Frozen phonons    -
ScanAxis            x [Å]             0.00 0.27 ... 3.81
ScanAxis            y [Å]             0.00 0.27 ... 3.81
RealSpaceAxis       x [Å]             0.00 0.04 ... 20.36
RealSpaceAxis       y [Å]             0.00 0.04 ... 20.36
```

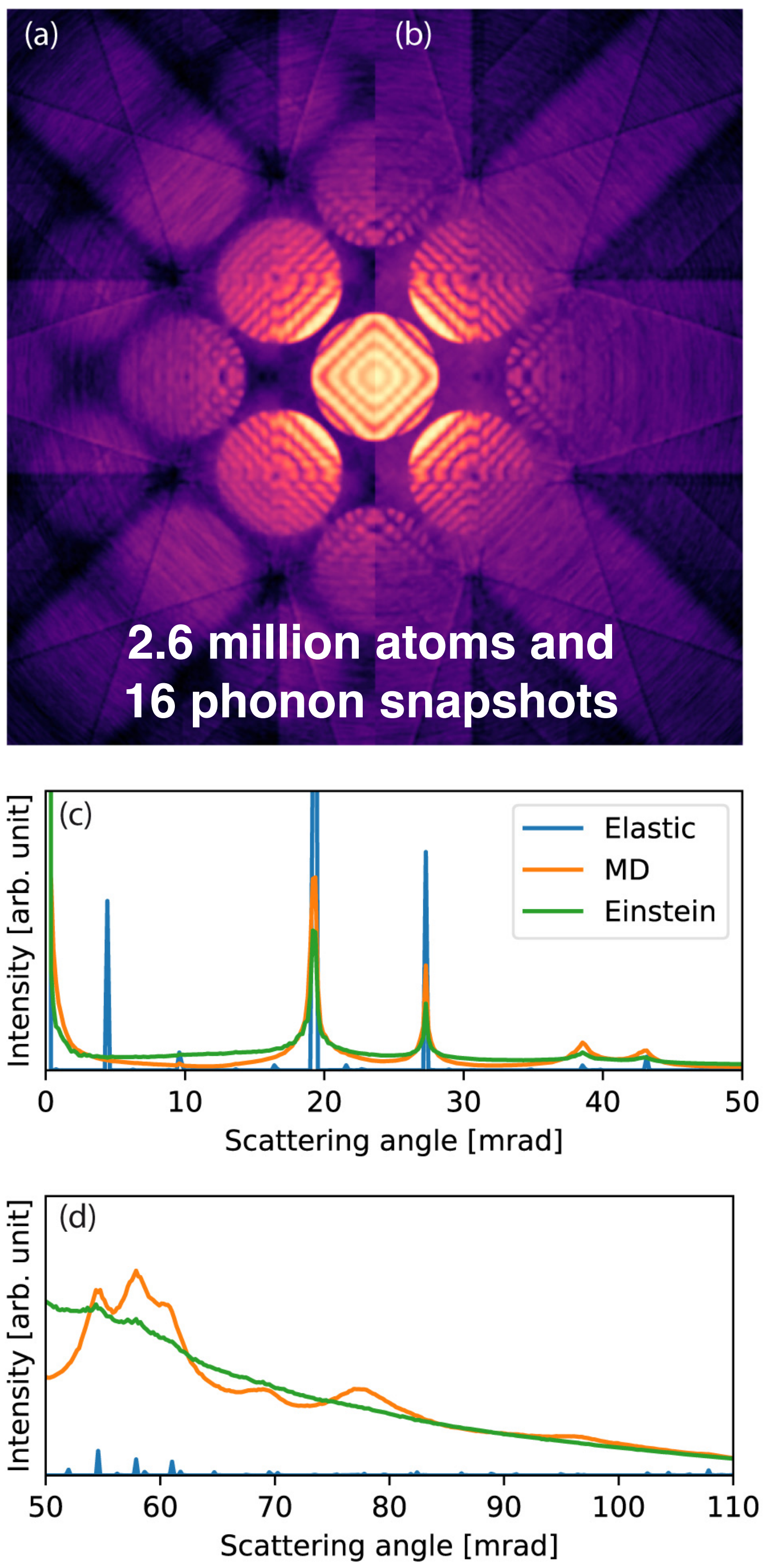
	Array	Chunk
Bytes	1.76 GiB	112.00 MiB
Shape	(4, 15, 15, 512, 512)	(1, 8, 7, 512, 512)
Count	17 Graph Layers	24 Chunks
Type	complex64	numpy.ndarray

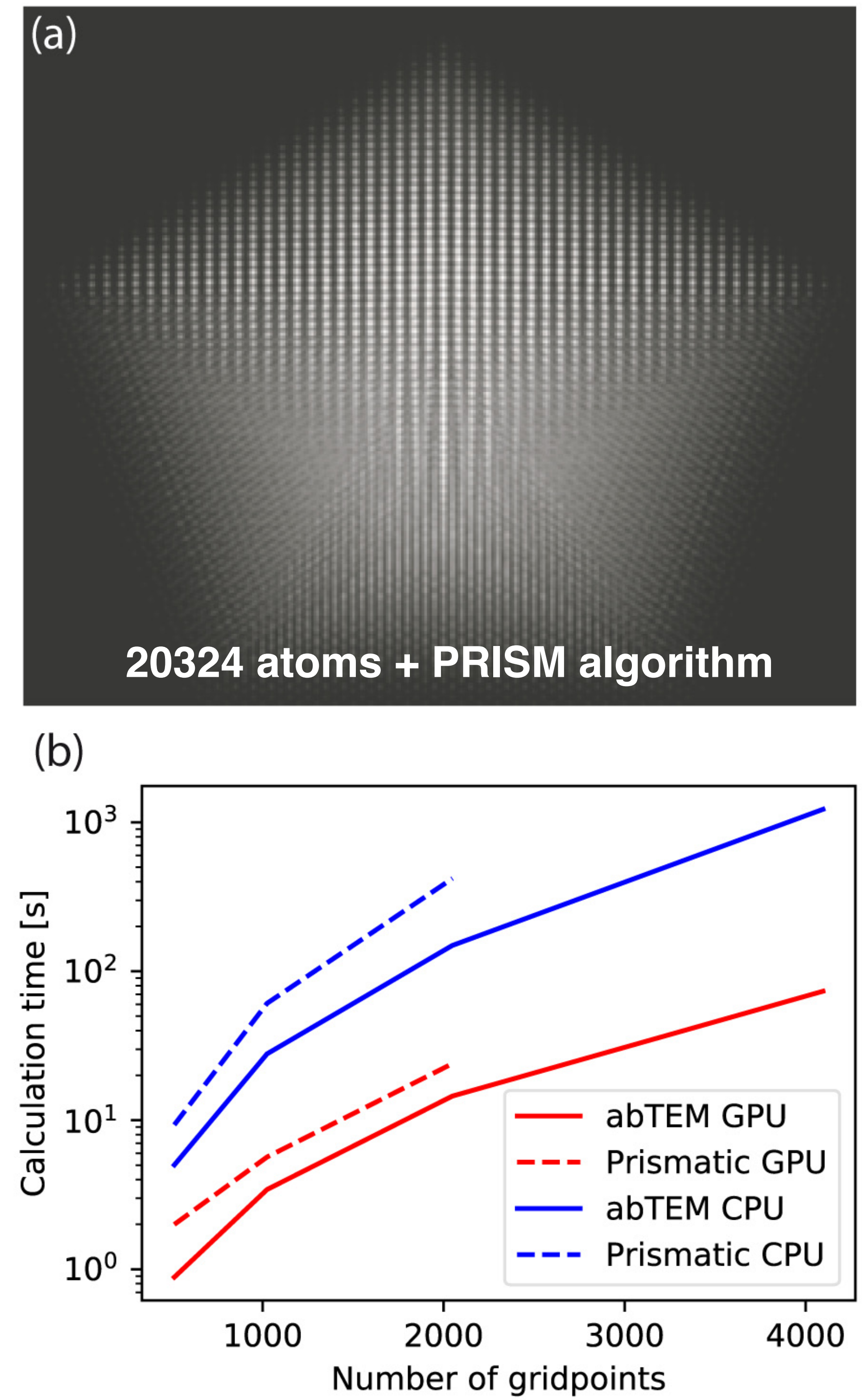




https://abtem.github.io/doc/user_guide/walkthrough/parallelization.html

*ab*TEM: Python, but *fast*





python™

jupyter

GPAW!

ASE

NumPy

CuPy

Numba

abTEM software tool article: J. Madsen & **TS**, Open Research Europe 1:24 (2021)
Fully functional open-source code: **https://github.com/abtem/abTEM**

*ab*TEM documentation

universität wien

*ab*TEM

Search ⌘ + K

Welcome!

Get started

- Overview
- Installation
- Basic examples

User guide

- Walkthrough
- Advanced tutorials
- Example gallery
- Bibliography

Reference

- API
- Configuration

abTEM

- Contact
- Contribute
- Cite
- Changelog

Welcome!

***ab*TEM: transmission electron microscopy from first principles**

*ab*TEM is a flexible open-source package for simulating transmission electron microscopy experiments. We strive for easy integration with other simulation codes and analysis tools accessible from Python. Utilization of other open-source projects makes *ab*TEM fast and scalable from a laptop to a high-performance computing environment.

Here are a few links to help you get started; please also note the navigation menu on the left side of the page.

Getting started

New to *ab*TEM? Visit our guides for getting started. They will introduce you to *ab*TEM's main concepts, help you install the code, and provide links to basic examples.

User guide

The user guide provides in-depth information about the key concepts of *ab*TEM with background information about simulations of transmission electron microscopy experiments.

Example gallery

Our example gallery provides fully functional code templates to assist with your common simulation tasks, for demonstrating a unique topic, or for reproducing result in published research articles.

API reference

Our reference contains a detailed description of the *ab*TEM API based on code documentation. It describes the methods and their parameters, and assumes that you have an understanding of the key concepts.

Next
Overview

https://abtem.github.io/doc