Monte Carlo simulation of phonon transport from ab-initio data with Nano-κ

B.H. Silva, David Lacroix, Mykola Isaiev, Laurent Chaput

Version of Record: https://www.sciencedirect.com/science/article/pii/S0010465523002990
Manuscript_b91bd4add366b16077ac277f9b252fb2

Monte Carlo simulation of phonon transport from *ab-initio* data with Nano-κ

B. H. Silva[a,*], D. Lacroix[a], M. Isaiev[a], L. Chaput[a,b]

[a] *Université de Lorraine, CNRS, LEMTA, Nancy F-54000, France*
[b] *Institut Universitaire de France, 1 rue Descartes, 75231 Paris, France*

Abstract

Understanding of heat transport in nanodevices is a challenging issue for several new technologies. It requires specific modelling approaches and dedicated simulation tools. Yet the latter are not numerous, and are often adapted to some "classic" materials or restricted to simple geometries (i.e. thin films, nanowires). Looking to address this deficiency, the present work brings Nano-κ, a Python code to simulate the phonon transport in nano and micro scale devices from *ab-initio* phonon data. The code has personalizing capabilities that allow to simulate several geometries and materials, offering insights of temperature distribution, heat flux, thermal conductivity, and mode contribution to energy transport. In the present work, a detailed description of the physical model implementation is provided and several simulation test cases are discussed. Nano-κ provides reliable predictions of the thermal conductivity in different materials, and is able to correctly predict the relative effect of size, temperature, and surface roughness on thermal transport properties.

Keywords: Monte Carlo, phonon transport, ab-initio, nanoscale heat transfer, thermal conductivity

PROGRAM SUMMARY

Program Title: Nano-κ
CPC Library link to program files: `https://doi.org/10.17632/tr29mhkjh8.1`
Developer's repository link: `https://github.com/brunohs1993/Nanokappa`
Licensing provisions: MIT

*Corresponding author
URL: `brunohartmann.silva@gmail.com` (B. H. Silva)

Programming language: Python
Nature of problem: The estimation of heat conduction at nanoscales depends on the estimation of phonon transport, since Fourier's law may not be valid in these situations. The phonon transport is described by the Boltzmann Transport Equation (BTE), that needs to be solved for each phonon mode separately, while estimating temperature distributions that depends on all modes at the same time. The phonon data can be retrieved from *ab-initio* simulations. Some BTE calculations have been already done through the years for simple geometries, but there is no publicly available code that allows enough flexibility to be used by the scientific community.
Solution method: Some previous works [1-3] have had success in applying the Monte Carlo method to solve the BTE for phonon transport. In the more recent works, the *ab-initio* data is retrieved from density functional theory (DFT) simulations [4-5]. Nano-κ is a Python code built on the same theoretical basis of previous developments, but adds more flexibility so that it can be open for public research use.

References

[1] David Lacroix, Karl Joulain, and Denis Lemonnier. Monte carlo transient phonon transport in silicon and germanium at nanoscales. Phys. Rev. B, 72:064305, Aug 2005. doi: 10.1103/PhysRevB.72.064305

[2] Laurent Chaput, Jérôme Larroque, Philippe Dollfus, Jérôme Saint-Martin, and David Lacroix. Ab initio based calculations of the thermal conductivity at the micron scale. Applied Physics Letters, 112(3):033104, 2018. doi: 10.1063/1.5010959

[3] Brice Davier, Jérôme Larroque, Philippe Dollfus, Laurent Chaput, Sebastian Volz, David Lacroix, and Jérôme Saint-Martin. Heat transfer in rough nanofilms and nanowires using Full Band Ab Initio Monte Carlo simulation. Journal of Physics: Condensed Matter, 30 (49):495902, October 2018. doi: 10.1088/1361-648X/aaea4f

[4] Atsushi Togo, Laurent Chaput, and Isao Tanaka. Distributions of phonon lifetimes in brillouin zones. Phys. Rev. B, 91:094306, Mar 2015. doi: 10.1103/PhysRevB.91.094306

[5] Atsushi Togo, Laurent Chaput, Terumasa Tadano, and Isao Tanaka. Implementation strategies in phonopy and phono3py. J. Phys. Condens. Matter, 35(35):353001, 2023. doi: 10.1088/1361-648X/acd831

1. Introduction

Energy efficiency is an important concern in almost every field involving energy conversion or energy transport. This is particularly true in nanoscale applications, especially in a world walking towards general digitalization, automation and downsizing. To fulfill this challenge, reliable tools are necessary to simulate and predict the thermal behaviour of nanodevices.

Nanoengineered devices and nanostructured materials have gained space in several fields, where different applications take advantage of their outstanding properties. Medicine, electronics, solar cells and waste heat management [5, 29, 19] are only a few examples. Understanding the thermal behaviour and operational limits of these devices ensures their safety and reliability, while preserving a compact design. The modelling of this thermal control, however, cannot be done by the usual macroscale approach, since Fourier's law may no longer be valid at scales smaller than a micrometer or for very low temperatures. In this case, the transport of phonons (quanta of lattice vibration) needs to be simulated instead of the total energy transport. The equation to be solved in this case is the Boltzmann transport equation (BTE), which describes the spatial-temporal distribution of the average number of phonons for each vibration mode of the crystal.

In order to simulate phonon transport in materials, one needs to know which frequencies are involved and consequently how much energy they can carry, as well as other related parameters. There are few methods able to predict the possible vibrational modes of a given crystalline material, for example based on the Density Functional Theory (DFT), as described by Togo et al. [27]. By knowing the atomic structure, the dispersion relations, phonon relaxation times and propagation velocities can be calculated.

To solve the BTE in nano and microstructures, the Monte Carlo method was used by Lacroix et al. [17], approximating the acoustic branches of Si and Ge using isotropic models and using semi-analytic lifetimes according to the M.G. Holland's formalism [14]. Then, in a more detailed approach, Chaput et al. [6] made estimations of thermal conductivity for cubic and hexagonal Si thin films with different thicknesses, also using the Monte Carlo method, this time combined with *ab-initio* data. Extending the latter work, Davier et al. [10] were able to model phonon transport also for rough nanofilms in the in-plane direction and across rough nanowires. Other approaches have also been implemented [26, 24].

The present work aims to further develop the method by using the com-

bination of Monte Carlo method and *ab-initio* data from previous research inside a framework that allows the user to simulate any geometry or semiconducting material. This flexibility aims to test new geometries and new materials to further improve and optimise nanodevices for relevant applications.

The computer code we describe in the following, Nano-κ, will be able to solve the Boltzmann equation, and compute heat fluxes, in systems with geometries ranging from tens of nanometers to microns. Of course, at very small length scale, the validity of our description, being semi-classical, must be considered carefully, since quantum effects will inevitably become important as the size is reduced. The limit of validity of the Boltzmann equation may often be found in the nanometer range.

2. Methodology

In this section we will describe the methodology implemented in Nano-κ, pronounced "nanokappa". Nano-κ is a computational code written in Python that aims to solve the Boltzmann Transport Equation (BTE) for phonons with the Monte Carlo method. The Nano-κ code is open source and available in its GitHub repository `https://github.com/brunohs1993/Nanokappa`. In the following subsections, the detailed implementation of the code and all related equations and models used as inputs and for the post-processing are discussed. The organization, its structure and the general flowchart are also presented. Additional information and examples can also be found in the GitHub repository.

2.1. Models and equations

The transport of phonons can be described mathematically by the Boltzmann Transport Equation (BTE) (Eq. 1). Each vibrational mode of the crystal has an associated wavevector $\mathbf{k}$ and branch (polarisation) index j. In this context, the BTE can be expressed for each mode as,

$$\frac{\partial n_{\mathbf{k}j}}{\partial t} + \nabla_{\mathbf{k}}\omega_{\mathbf{k}j} \cdot \nabla_{\mathbf{r}} n_{\mathbf{k}j} = \left.\frac{\partial n_{\mathbf{k}j}}{\partial t}\right|_{scat}, \tag{1}$$

where the occupation number $n_{\mathbf{k}j} = n_{\mathbf{k}j}(\mathbf{r}, t)$ is the average number of phonons in mode $(\mathbf{k}j)$. The number of phonons, at time t, in mode $(\mathbf{k}j)$, in a small volume $d^3\mathbf{r}$ around point $\mathbf{r}$, with wavevector in the range $d^3\mathbf{k}$ around

$\mathbf{k}$ is $n_{\mathbf{k}j} d^3\mathbf{r} d^3\mathbf{k}/(2\pi)^3$. ω is the angular frequency, and the sub-index *scat* refers to the variation of n due to the scattering of the phonons in mode $(\mathbf{k}j)$ with other phonons (phonon-phonon interactions), or due to other scattering mechanisms. The symbol $\nabla_{\mathbf{k}}$ refers to the gradient applied in the reciprocal space, while $\nabla_{\mathbf{r}}$ is the gradient in the real space.

The first term in Eq. 1 refers to the local variation of the number of phonons in time. The gradient of ω in the reciprocal space $\mathbf{k}$ is equal to the group velocity $\mathbf{v}$ of that mode,

$$\mathbf{v}_{\mathbf{k}j} = \nabla_{\mathbf{k}} \omega_{\mathbf{k}j}, \tag{2}$$

which makes the second term the balance of phonon flux due to the phonons drifting through space. The term on the right hand side of the BTE refers to the variation due to collisions between phonons. In this work, we only consider phonon-phonon interactions for the scattering, thus transferring energy between modes due to inherent anharmonicities in the inter-atomic potentials. According to the relaxation time approximation [6], the phonon-phonon scattering can be modelled as,

$$\left.\frac{\partial n_{\mathbf{k}j}}{\partial t}\right|_{scat} = \frac{n^0_{\mathbf{k}j}(T) - n_{\mathbf{k}j}}{\tau_{\mathbf{k}j}(T)}, \tag{3}$$

where τ is the relaxation time of the mode and n^0 is the number of phonons of that mode at equilibrium, both of them function of local temperature T.

The number of phonons at equilibrium $n^0_{\mathbf{k}j}(T)$ is given by the Bose-Einstein distribution,

$$n^0_{\mathbf{k}j}(T) = \frac{1}{\exp[\hbar\omega_{\mathbf{k}j}/k_b T] - 1}, \tag{4}$$

where $\hbar$ is the reduced Planck's constant and k_b is the Boltzmann constant.

The temperature T is defined at equilibrium. However, a local temperature $T(\mathbf{r}, t)$ can be defined assuming a local equilibrium for the occupation function, and mimicking the equilibrium theory.

At equilibrium, the energy density e is computed summing the contributions from all phonon modes. If V is the volume of the crystal, N the number of primitive unit cells of volume V_0 in the crystal, containing each N_a atoms,

then

$$e = \sum_{j=1}^{3N_a} \int_{BZ} \hbar\omega_{\mathbf{k}j} \left[n^0_{\mathbf{k}j}(T) + \frac{1}{2} \right] \frac{d^3\mathbf{k}}{(2\pi)^3} \tag{5a}$$

$$= \frac{1}{V_0 N} \sum_{\mathbf{k}j} \hbar\omega_{\mathbf{k}j} \left[n^0_{\mathbf{k}j}(T) + \frac{1}{2} \right] \tag{5b}$$

The BZ subtext indicates an integration performed over the first Brillouin Zone.

Eq. 5b is a relation of the form $e = f(T)$. Inverting this relation, $T = f^{-1}(e)$, the temperature can be obtained for a given energy density. In a local equilibrium state, this equation can be used locally to compute a local temperature. Indeed, for a local equilibrium occupation $n_{\mathbf{k}j}$, we compute the energy density from

$$e = \sum_{j=1}^{3N_a} \int_{BZ} \hbar\omega_{\mathbf{k}j} \left[n_{\mathbf{k}j} + \frac{1}{2} \right] \frac{d^3\mathbf{k}}{(2\pi)^3}, \tag{6}$$

and obtain the temperature as $T = f^{-1}(e)$, where f^{-1} is the function obtained for the system in equilibrium.

Practically, during computations, the number of primitive cells in the crystal, N, which equal the number of wavevectors $\mathbf{k}$ within the first Brillouin zone, cannot be taken to be infinite. Thus, a large finite number is chosen, and as a consequence the number of wavevectors within the first Brillouin zone become finite. Those vectors are denoted as

$$\mathbf{k}_i, \quad i = 1 \cdots N. \tag{7}$$

The energy density is then computed as

$$e = \frac{1}{NV_0} \sum_{j=1}^{3N_a} \sum_{i=1}^{N} \hbar\omega_{\mathbf{k}_i j} \left[n_{\mathbf{k}_i j} + \frac{1}{2} \right]. \tag{8}$$

2.2. Input Ab-initio data

Our calculations use as input data the phonon frequency, group velocities and lifetimes obtained from *ab-initio* calculations.

The phonon frequencies and group velocities, $\omega_{\mathbf{k}j}$ and $\mathbf{v}_{\mathbf{k}j}$, can be computed from lattice dynamics [23], knowing the harmonic part of the interaction potential between atoms. The phonon lifetime $\tau_{\mathbf{k}j}$ is calculated with the many-body perturbation theory [22], knowing the anharmonic part of the interaction potential. The harmonic and anharmonic part of the interaction potential are represented by harmonic and anharmonic force constants, which are the second and third derivative of the interaction energy with respect to the atomic displacements. This interaction energy can be computed quantum mechanically within the framework of Density Functional Theory (DFT), and the derivatives can be performed numerically from finite displacements, by moving the atoms around their equilibrium positions. This strategy has recently been implemented within the Phono3py computer code [27]. For a given bulk material, the results of such calculations are stored in a HDF5 file, which contains the $\mathbf{v}$, ω and $\tau(T)$ values for each combination of the band index j, and wavevector $\mathbf{k}$ within the irreducible part of the first Brillouin zone [28]. Those data are used as input for our calculations.

2.3. Geometry setting

The geometry is loaded with the help of the Trimesh Python module [11]. It can be input as a triangular mesh from an STL file or created with a keyword accepted by Nano-κ, such as `box`, `cylinder` or `zigzag`, along with the corresponding parameters (length, radius, etc.). If the user wishes to generate its own STL file to be loaded, it can be done by using Nano-κ's Mesh class capabilities. A Mesh object can be created from a list of vertices and faces, pre-processed, visualised and exported as an ASCII STL file. Alternatively, any external software such as FreeCAD [16] or Blender [8] can be used to generate the desired mesh. Figure 1 shows the different results for different geometric input parameters.

In Nano-κ, a triangle defined by three vertices is called a "face", while a group of adjacent and coplanar triangles is called a "facet". This nomenclature is inherited from Trimesh vernacular, despite Nano-κ using its own dedicated Mesh class in the calculation. One important difference to notice is that Trimesh does not consider facets composed of one single triangle, while Nano-κ Mesh object does.

The main mesh is imported and processed to get all relevant geometric properties (normals, surface areas, edges, boundaries, etc.). In order to be consistent with the units from the *ab-initio* data, all the positions in the mesh are assumed to be expressed in angstroms [Å], the standard unit of length

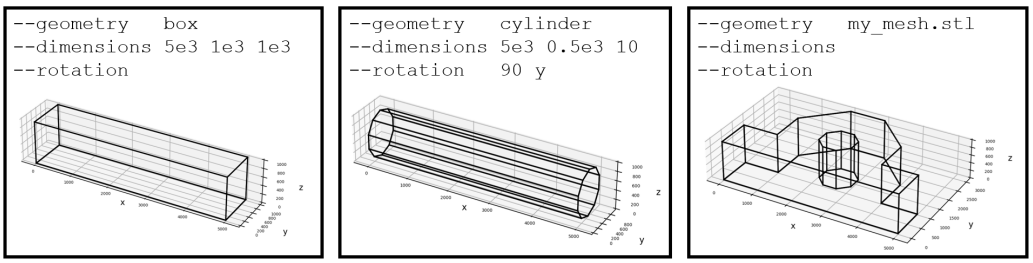

Figure 1: Examples of geometries that can be simulated: box, cylinder (with custom number of sides), and custom geometries.

of the code. To minimize possible numerical errors, all vertex positions are rounded to the 10^{-1} Å. Given that the scale of the simulations go from the order of the tens of nanometers (10 nm = 10^2 Å) to tens of micrometers (10 μm = 10^5 Å) the effect of this rounding on the results is considered as insignificant.

After the mesh is processed, the boundary conditions (BC) are assigned. The BC are imposed to a whole facet of the geometry. The current code supports three types of boundary conditions:

- **T - Temperature:** Constant temperature on the facet. This is done by assuming that the facet is in contact with a thermal reservoir, at equilibrium, at the prescribed temperature, i.e. a black body. The phonon population inside the reservoir is therefore distributed according to the Bose-Einstein distribution, Eq. 4. These phonons are injected from the reservoir into the domain throughout the simulation. The rate of injection is function of their group velocity and direction in relation to the normal of the facet. Every particle that leaves the domain into a reservoir is completely absorbed and has no effect on the reservoir's temperature. See Section 2.4.2 for a detailed expression of the injection rate.

- **P - Periodic:** Periodic boundary condition allows to simulate objects which have a periodicity. Examples are shown in Fig. 2, where a *period* is repeated in 1 or 2 directions (Nano-κ accepts periodicities in all 3 directions). For the BC to be properly set, the equivalent facets of the period need to have parallel normals pointing in opposite directions. The phonons that leave the period through one of these facets are then translated to the respective equivalent one, keeping the same position

relative to the vertices of the facet. It is important to note that orientation matters because modes properties can change with the orientation, therefore the restriction on parallel normals.

- **R - Roughness:** a roughness can be imposed to the boundaries of the geometry to model the reflection of phonons back to the domain. The measure of roughness used in Nano-κ is defined by Ziman [30] by assuming that the deviation of any point on the surface from its mean plane obeys a normal distribution. The roughness η can then be taken as standard deviation of this distribution. Figure 3 exemplifies the definition. For the Ziman model, the higher the roughness, the more diffuse are the reflections, as it will be seen in Section 2.4.5.

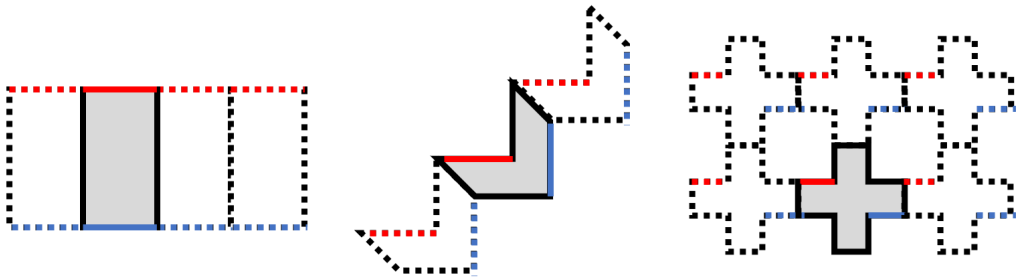

Figure 2: Examples of periodic geometries along one or two directions. Nano-κ accepts periodicity in all 3 directions. The red and blue lines represent the imposition of hot and cold temperatures, respectively. All the remaining non-periodic faces should have an associated roughness as BC.

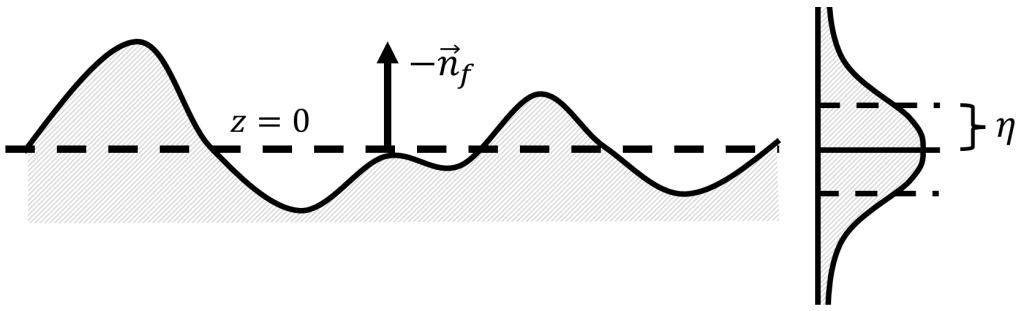

Figure 3: Illustration of the definition of roughness η as described by Ziman [30].

The last part of the geometry processing is to divide the domain into subvolumes, such that it is possible to estimate the local temperature within each region in the domain. The code uses a method of "reference points" in order to define in which subvolume each of the particles are contained. These "reference points" are points distributed in the bounding box of the geometry in different ways according to the subvolume type defined by the user (`slice`, `grid` or `voronoi`), which will be described next. Each reference point represents a subvolume and is indexed as subvolume 0, 1, 2, etc. A particle in the domain is classified to a given subvolume based on its proximity to the reference points. For instance, if in a given moment a particle is closest to the reference point 2, it will be classified as being contained in subvolume 2, and its energy will be used to calculate T in that region. Over time, the particle will drift and approach another reference point, transporting its energy to other subvolume. To do that efficiently, a nearest-neighbour interpolator is defined, using the position of the particles as input and returning their respective subvolume indices.

Three types of subvolume can be set:

- **Slice:** equidistant slicing of the domain along a given direction (x, y or z). This is useful to calculate thermal conductivity of simple geometries according one dimensional settings, such as thin films and nanowires;

- **Grid:** consecutive slicing in one, two or three directions. Depending of the geometry, some of the resulting reference points will be located out of the domain or will be isolated from other reference points. These are removed automatically. This type of subdivision can be used to detect wall effects or to properly divide a simple geometry with multi-directional heat flux, such as a T-shaped or cross-shaped geometry.

- **Voronoi:** for more complex geometries (with holes, inclusions, sharp corners), when none of the previous options are suitable, there is a third option that iteratively creates a Voronoi diagram of the geometry. The division is done the following way:

    1. Uniformly sample the geometry's volume, with samples positions $\mathbf{r}_s$;
    2. Generate random reference points for the subvolumes, with positions $\mathbf{r}_{sv}$;
    3. Classify each $\mathbf{r}_s$ according to the nearest $\mathbf{r}_{sv}$;

4. Update each $\mathbf{r}_{sv}$ to the mean position of the correspondent $\mathbf{r}_s$ (in other words, the current "center of mass" of each subvolume);
5. If there is no change, refine the volume sampling by increasing the number of samples;
6. When the number of samples is at the maximum allowed value and there is no change when updating $\mathbf{r}_{sv}$, the subvolumes are set.

Figure 4 exemplifies in 2D what the algorithm does in three dimensions.

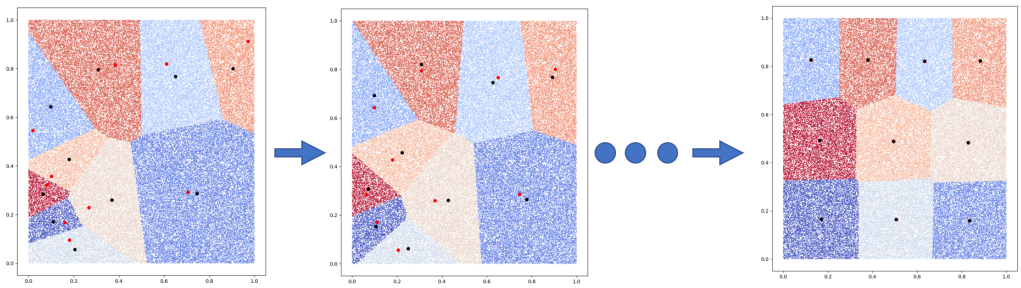

Figure 4: Two-dimensional exemplification of the algorithm to set the Voronoï subvolumes. The red dots represent the current $\mathbf{r}_{sv}$, which will be updated to the black dots for the next iteration until there is no more change (on the right).

Once the geometry processing is done, the geometry and the material data are combined in the Monte Carlo approach.

2.4. Monte Carlo approach

The Monte Carlo simulation calculates the time evolution of particles through the domain defined by the user. In Nano-ĸ, each particle represents a wave-packet associated to a vibrational mode ($\mathbf{k}j$), as described in [6]. Consequently, each particle in the simulation carries information about the current state of that vibrational mode at the position it finds itself located: frequency, number of phonons, local temperature, etc.

2.4.1. Initialisation

At the beginning of the simulation, the system is initialised from an initial temperature profile and the phonon distribution is computed using the Bose-Einstein distribution (Eq. 4). The assumed temperature profile can be set

as a step function, an uniform temperature, a linear profile or customised by the users as they wish.

Once the initialization is done, the phonon distribution evolves in time according to the Boltzmann equation (Eq. 1). It is however not directly the dynamics of phonons that we simulate during our calculations but rather the dynamics of particles. Indeed, to every mode $\mathbf{k}j$, we associate N_{ps} wave-packets per subvolume, as shown by Chaput et al. [6], and propagate them along the characteristic lines of the Boltzmann equation. Those wave-packets contains the phonons. When a single wave-packet is used to represent a phonon mode $\mathbf{k}j$, considering a volume V_s at the equilibrium temperature T, it is shown by Chaput et al. [6] that the number of phonons contained in this wave-packet is $\frac{V_s}{V_0 N} n^0_{\mathbf{k}j}(T)$. Therefore if we use N_{ps} wave-packets to represent that mode, each wave-packet will contain $\frac{V_s}{V_0 N N_{ps}} n^0_{\mathbf{k}j}(T)$ phonons. Increasing N_{ps} allows to reach better statistics, naturally it has a detrimental effect on simulation time as more particles have to be considered.

2.4.2. Injection of particles

To impose a constant temperature T on a facet, phonons are injected into the domain from a reservoir at all times. The wall emits and absorbs phonons as a black body. Any phonon reaching the facet is absorbed, and phonons are emitted according to the Bose-Einstein distribution. The number of phonons, per unit volume at a temperature T is $\frac{1}{V_0 N} \sum_{\mathbf{k}j} n^0_{\mathbf{k}j}(T_f)$. Therefore the number of phonon entering the volume during the time Δt from a facet with surface A and unit outwards normal vector $\mathbf{n}$ is

$$N(T) = \frac{A\Delta t}{V_0 N} \sum_{\mathbf{k}j} n^0_{\mathbf{k}j}(T) |\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}| \theta(-\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}) \tag{9}$$

where θ is the Heaviside step function. The number of particles of mode $\mathbf{k}j$ to be injected though the surface A during the time Δt is therefore

$$\begin{aligned} R_{\mathbf{k}j} &= A\Delta t \frac{N_{ps}}{V_s} |\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}| \theta(-\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}) \\ &= A\Delta t \frac{\rho}{3 N_a N} |\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}| \theta(-\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}) \end{aligned} \tag{10}$$

where V_s is the volume of the subvolume containing the facet and $\rho = \frac{3 N_a N N_{ps}}{V_s}$ the density of particles in that subvolume. We can also compute the density of

particle from the total number of particles in the system, $N_p = 3N_aN\sum_s N_{ps}$, and the total volume, $V = \sum_s V_s$, $\rho = \frac{N_p}{V}$.

The quantity $R_{\mathbf{k}j}$ indicates how many particles of mode $\mathbf{k}j$ have to be injected every timestep. For example, if $R_{\mathbf{k}j} = 2.5$ particles/timestep, it means that every timestep 2 particles of mode $\mathbf{k}j$ are necessarily added to the domain, and an additional third particle have a 50% chance of being generated. This is calculated by comparing a random number generated from a uniform distribution to the non-integer part of $R_{\mathbf{k}j}$. If it the random number is smaller, the additional particle is added to the system.

After the entering modes are defined the incoming particles are randomly generated at the facet of the domain. A random number in the interval [0, 1) defines how much of the timestep the particle has drifted before entering the geometry. This ensures a truly random distribution by not letting every new particle start from the same point, hence emulating an external black body.

2.4.3. Drifting and local temperature

Every particle drifts through the domain with the group velocity of its corresponding mode. As shown by Chaput et al. [6], along the characteristic lines of the Boltzmann equation, at each iteration, the position $\mathbf{r}_i$ of a particle i is updated according to

$$\mathbf{r}_i^k = \mathbf{r}_i^{k-1} + \mathbf{v}_i \Delta t, \quad i = 1, \ldots, N_p, \tag{11}$$

where $\mathbf{r}_i^k$ is the position of particle i at timestep k, and $\mathbf{v}_i$ is the group velocity of that particle. After drifting, some particles will have changed of subvolumes, transferring energy between them. This will cause a variation on the local energy density, and consequently on the local temperature T_s of the subvolume s. The local energy density in subvolume s is calculated using Eq. 5b

$$\begin{aligned} e_s &= \frac{1}{V_0 N} \sum_{\mathbf{k}j} \hbar\omega_{\mathbf{k}j} \left[n^0_{\mathbf{k}j}(T_{ref}) + \frac{1}{2} \right] + \frac{1}{V_0 N} \sum_{\mathbf{k}j} \hbar\omega_{\mathbf{k}j} \delta n_{\mathbf{k}j}(T_s) \\ &= e_{ref} + \frac{1}{V_s} \sum_{\mathbf{k}j} \sum_{p=1}^{N_{ps}} \hbar\omega_{\mathbf{k}j} \left(V_s \frac{\delta n_{\mathbf{k}j}(T_s)}{V_0 N N_{ps}} \right) \\ &= e_{ref} + \frac{1}{V_s} \sum_{i=1}^{3N_aNN_{ps}} \hbar\omega_i N_i \end{aligned} \tag{12}$$

where $\hbar\omega_i$ is the energy of the wave-packet i, and N_i the number of phonons in this wave-packet, above the reference temperature,

$$N_i = V_s \frac{\delta n_{\mathbf{k}j}(T_s)}{V_0 N N_{ps}}. \tag{13}$$

In this equation, the number of particles per mode, N_{ps}, may or may not be given directly as input by the user. If it is not given, the total number of particles in the system is known and N_{ps} is computed as $N_{ps} = \frac{N_{p,sv}}{3N_a N}$, where $N_{p,sv}$ is the instantaneous number of particles in the subsystem.

The total energy summation uses a reference temperature T_{ref}, with a corresponding reference energy density e_{ref}. This reference energy density can be calculated from a fixed T value (e.g. 300 K) or using the local temperature T_s, varying according to the position of the particle. This approach considers that the local energy variation depends on the deviation of each mode's occupation from T_{ref}:

$$\delta n_{\mathbf{k}j} = n_{\mathbf{k}j} - n^0_{\mathbf{k}j}(T_{ref}) \tag{14}$$

This helps the Monte Carlo calculations to have better statistics due to its random nature and decreased population. Using the absolute value can cause a high variation depending on the number of particles, and a very large number of particles would require an infeasible amount of computational time. The closer T_{ref} is from the local temperature, smaller is δn and more stable are the calculations. With the local energy density, the local temperature T_s can be estimated.

2.4.4. Time convergence of the phonon population

As shown by Chaput et al. [6], along the characteristic lines $\mathbf{r}(t) = \mathbf{r}(t_0) + \mathbf{v}_{\mathbf{k}j}(t - t_0)$ the Boltzmann equation can be written as

$$\frac{dn_{\mathbf{k}j}(t)}{dt} + \frac{1}{\tau_{\mathbf{k}j}(t)} n_{\mathbf{k}j}(t) = \frac{n^0_{\mathbf{k}j}(t)}{\tau_{\mathbf{k}j}(t)}. \tag{15}$$

In the above equation the time dependence of each factor has been made explicit. In the Boltzmann equation, the phonon population is $n_{\mathbf{k}j}(\mathbf{r}, t)$, but along the characteristic lines it becomes $n_{\mathbf{k}j}(\mathbf{r}(t), t) \equiv n_{\mathbf{k}j}(t)$. The time dependence of the Bose-Einstein distribution and lifetime are obtained indirectly through the temperature, $n^0_{\mathbf{k}j}(T(\mathbf{r}(t))) \equiv n^0_{\mathbf{k}j}(t)$ and $\tau_{\mathbf{k}j}(T(\mathbf{r}(t))) \equiv \tau_{\mathbf{k}j}(t)$.

To solve Eq. 15 numerically, the time derivative is usually approximated using first order finite differences considering a small time step Δt after a current time t_0, such as we get

$$\frac{n_{\mathbf{k}j}(t_0+\Delta t)-n_{\mathbf{k}j}(t_0)}{\delta t}+\frac{1}{\tau_{\mathbf{k}j}(t_0)}n_{\mathbf{k}j}(t_0)=\frac{n^0_{\mathbf{k}j}(t_0)}{\tau_{\mathbf{k}j}(t_0)}, \tag{16}$$

or

$$n_{\mathbf{k}j}(t_0+\Delta t)=n_{\mathbf{k}j}(t_0)+\frac{\Delta t}{\tau_{\mathbf{k}j}(t_0)}(n^0_{\mathbf{k}j}(t_0)-n_{\mathbf{k}j}(t_0)). \tag{17}$$

This equation allows to evaluate the phonon population at the next time step. However it assumes that $n^0_{\mathbf{k}j}$ and $\tau_{\mathbf{k}j}$ have little variation in one timestep, virtually assuming a constant temperature inside this interval. This method works well if $\Delta t \ll \tau_{\mathbf{k}j}$, but can lead to instability problems as $\Delta t \to 2\tau_{\mathbf{k}j}$, diverging the calculations beyond this limit. It clearly imposes a cap on the timestep that can be chosen by the user even if they wish to do quick approximate calculations and are not concerned with high precision.

The change of the phonon population in a time step Δt can also be obtained after having integrated the Boltzmann equation. Indeed, Eq. 15 can be solved using the integrating factor

$$\mu_{\mathbf{k}j}(t)=e^{\int^t \frac{ds}{\tau_{\mathbf{k}j}(s)}} \Longrightarrow \frac{d\mu_{\mathbf{k}j}(t)}{dt}=\frac{\mu_{\mathbf{k}j}(t)}{\tau_{\mathbf{k}j}(t)}. \tag{18}$$

We obtain

$$\mu_{\mathbf{k}j}(t)\frac{dn_{\mathbf{k}j}(t)}{dt}+\frac{\mu_{\mathbf{k}j}(t)}{\tau_{\mathbf{k}j}(t)}n_{\mathbf{k}j}(t)=\mu_{\mathbf{k}j}(t)\frac{n^0_{\mathbf{k}j}(t)}{\tau_{\mathbf{k}j}(t)} \tag{19}$$

or, integrating between t_0 and $t_0+\Delta t$,

$$n_{\mathbf{k}j}(t_0+\Delta t)=\frac{\mu_{\mathbf{k}j}(t_0)}{\mu_{\mathbf{k}j}(t_0+\Delta t)}n_{\mathbf{k}j}(t_0)+\int_{t_0}^{t_0+\Delta t}\frac{\mu_{\mathbf{k}j}(t)}{\mu_{\mathbf{k}j}(t_0+\Delta t)}\frac{n^0_{\mathbf{k}j}(t)}{\tau_{\mathbf{k}j}(t)}. \tag{20}$$

To obtain Eq. 17, the lifetime and the Bose-Einstein distribution were assumed to be constant in the interval between t_0 and $t_0+\Delta t$. If we use the same approximation here, then $\mu_{\mathbf{k}j}(t)=e^{\frac{t-t_0}{\tau_{\mathbf{k}j}(t_0)}}$ and we obtain

$$n_{\mathbf{k}j}(t_0+\Delta t)=n_{\mathbf{k}j}(t_0)e^{-\frac{\Delta t}{\tau_{\mathbf{k}j}(t_0)}}+n^0_{\mathbf{k}j}(t_0)(1-e^{-\frac{\Delta t}{\tau_{\mathbf{k}j}(t_0)}}). \tag{21}$$

In the limit of small time step, Eq. 17 is recovered from Eq. 21 at linear order in Δt. Eq. 21 is the only discretisation obtained from the integrating factor method we have tested, but Eq. 20 allows, in principle, to obtain better ones.

Figure 5 shows the comparison between the schemes given by Eq. 17 and Eq. 21. In order to compare them, an arbitrary linear profile of n^0 was chosen. This profile can approximate, in fairly high temperatures, the n^0 profile a particle would encounter in a linearly decaying temperature field. The relaxation time τ is considered to be constant. The initial occupation of the particle, $n(t = 0)$, was arbitrarily set to 2, and will approximate n^0 in the long time limit. To calculate this evolution, each plot applies the linear and the exponential schemes with different values of τ, for a fixed $dt = 1$. It is clear that, for small values of dt/τ, both methods are almost equivalent, becoming equal as $dt/\tau \to 0$. When $dt/\tau > 1$, instabilities start to arise with the linear scheme, and from $dt/\tau \geq 2$, they are unsustainable, leading to divergence. The exponential scheme, on the contrary, keeps itself stable despite the increase of dt/τ. It is also important to note that the results also vary with timestep size for a fixed τ. The smaller is dt, the closer are both schemes. A smaller timestep also helps to stabilise the calculation, but increases the time needed. The effects of both schemes on the thermal conductivity will be discussed in Section 3.1.4.

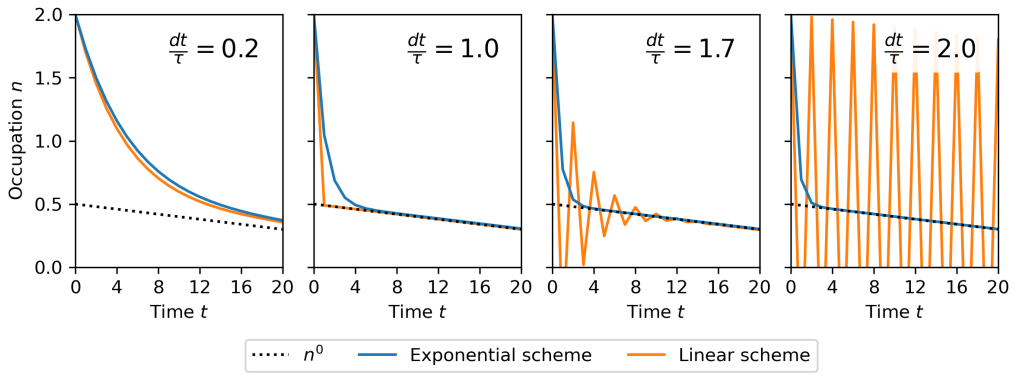

Figure 5: Comparison of the linear scheme used in previous works with the exponential scheme used in the present study.

2.4.5. Boundary scattering

When a phonon is scattered by a surface, it undergoes a transition from a mode $\mathbf{k}j$ to a mode $\mathbf{k}'j'$. Quantum mechanically, the probability of this transition could be computed using the Fermi golden rule, or from the t-matrix, the boundary being seen as a perturbation. However, in our approach, phonons are modelled using wave-packets, and therefore scattering on surfaces should also be described based on those quantities. A common approach is to introduce specular scattering. In such process, a particle approaches the surface in a mode $\mathbf{k}j$, is scattered, and then emerges from the surface in a mode $\mathbf{k}'j'$ fulfilling the equations

$$\omega_{\mathbf{k}'j'} = \omega_{\mathbf{k}j}, \tag{22}$$

$$\mathbf{v}_{\mathbf{k}'j'} = \mathbf{v}_{\mathbf{k}j} - 2(\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n})\mathbf{n}, \tag{23}$$

where $\mathbf{n}$ is the outward normal unit vector to the surface. In such process the energy is conserved, and the velocity vector is reflected. We assume there is only one such mode $\mathbf{k}'j'$ to fulfil those conditions. A transition which is not specular is called diffuse.

Based on this definition, we define a scattering model to compute the probability $P(\mathbf{k}j \to \mathbf{k}'j')$ for a particle to make a transition from mode $\mathbf{k}j$ to mode $\mathbf{k}'j'$. This particle can reach the mode $\mathbf{k}'j'$ though a specular or a diffuse transition, therefore

$$P(\mathbf{k}j \to \mathbf{k}'j') = P^S(\mathbf{k}j \to \mathbf{k}'j') + P^D(\mathbf{k}j \to \mathbf{k}'j'), \tag{24}$$

where $P^S(\mathbf{k}j \to \mathbf{k}'j')$ is the probability for a particle in mode $\mathbf{k}j$ to make a specular transition to mode $\mathbf{k}'j'$ and $P^D(\mathbf{k}j \to \mathbf{k}'j')$ is the probability for a particle in mode $\mathbf{k}j$ to reach that mode through a diffuse scattering.

If we call $N^>_{\mathbf{k}j}$ the number of particles in mode $\mathbf{k}j$ which will be scattered by the surface A during Δt,

$$N^>_{\mathbf{k}j} = \rho_m A \Delta t (\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}) \theta(\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}), \tag{25}$$

where ρ_m is the density of particle per mode, assumed to be the same for all modes, and $N^{>S}_{\mathbf{k}j}$ the fraction of those particles for which the scattering will be specular, we have

$$P^S(\mathbf{k}j \to \mathbf{k}'j') = \frac{N^{>S}_{\mathbf{k}j}}{N^>_{\mathbf{k}j}} \equiv p_{\mathbf{k}j}. \tag{26}$$

Here and in the following we use the superscript $>$ to refer to quantities computed before the scattering, because in that case the velocity of the particles fulfill $\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n} > 0$. For quantities computed after the scattering on the surface we use $<$ because then $\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n} < 0$. The above equation defines the specularity of the initial mode, $p_{\mathbf{k}j}$, which is a parameter of the model. In the limit of long wavelengths, Ziman [30] has shown that $p_{\mathbf{k}j}$ could be expressed using the standard deviation η of the surface from a reference plane

$$p_{\mathbf{k}j} = \exp[-(2\|\mathbf{k}\|\eta\cos\theta_{\mathbf{k}j})^2], \tag{27}$$

where $\theta_{\mathbf{k}j}$ is the angle of incidence of the particle on the facet,

$$\cos\theta_{\mathbf{k}j} = \frac{\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}}{\|\mathbf{v}_{\mathbf{k}j}\|}. \tag{28}$$

The number of particles in mode $\mathbf{k}j$ for which the scattering will be diffuse is $N^{>D}_{\mathbf{k}j} = N^{>}_{\mathbf{k}j} - N^{>S}_{\mathbf{k}j}$, and we call $N^{<D}_{\mathbf{k}'j'}$ the number of particles that end up in mode $\mathbf{k}'j'$ from a diffuse scattering from any possible incoming mode. $N^{<D} = \sum_{\mathbf{k}'j'} N^{<D}_{\mathbf{k}'j'}$ is therefore the total number of particles that had a diffuse scattering. From those definitions we obtain

$$P^D(\mathbf{k}j \to \mathbf{k}'j') = (1 - p_{\mathbf{k}j})\frac{N^{<D}_{\mathbf{k}'j'}}{N^{<D}}. \tag{29}$$

$N^{<D}_{\mathbf{k}'j'}$ is obtained from the following argument: the number of particles before and after the collision is the same, $N^{>} = N^{>S} + N^{>D} = N^{<} = N^{<S} + N^{<D}$. Moreover, the particle specularly scattered are in one to one correspondence, through Eqs. 22 and 23, before and after the collision with the surface, therefore $N^{>S}_{\mathbf{k}j} = N^{<S}_{\mathbf{k}'j'}$ and $N^{>S} = N^{<S}$. This gives $N^{>D} = N^{<D}$, the number of particles to have diffused scattering is the same before and after the collision. $N^{>D}$ is easily computed as

$$N^{>D} = \sum_{\mathbf{k}j} N^{>D}_{\mathbf{k}j} = \sum_{\mathbf{k}j} (1 - p_{\mathbf{k}j}) N^{>}_{\mathbf{k}j} = \sum_{\mathbf{k}j} (1 - p_{\mathbf{k}j}) \rho_m A \Delta t \mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n} \theta(\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}). \tag{30}$$

In the above equation each mode $\mathbf{k}j$, fulfilling $\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n} > 0$, is in one to one correspondence with its specular image denoted $\mathbf{k}'j'$, for which Eq. 23 gives $\mathbf{v}_{\mathbf{k}'j'} \cdot \mathbf{n} = -\mathbf{v}_{\mathbf{k}j} \cdot \mathbf{n}$. Notice that the conservation of the number of scattered particles, $N^{>S}_{\mathbf{k}j} = N^{<S}_{\mathbf{k}'j'}$, impose for Eq. 23 to be written in term of the group velocity, and not wavevectors, as it is sometimes done.

We obtain

$$N^{>D} = -\sum_{\mathbf{k}'j'}(1-p_{\mathbf{k}j})\rho_m A\Delta t \mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n}\theta(-\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n}) \tag{31}$$

and equating this with $N^{<D} = \sum_{\mathbf{k}'j'} N^{<D}_{\mathbf{k}'j'}$ suggest the identification

$$N^{<D}_{\mathbf{k}'j'} = -(1-p_{\mathbf{k}j})\rho_m A\Delta t \mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n}\theta(-\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n}). \tag{32}$$

This equation refer to the state $\mathbf{k}j$ through $p_{\mathbf{k}j}$. Therefore even if particles are diffusely scattered into state $\mathbf{k}'j'$, they are nevertheless seen as the specular image of particles with an initial mode $\mathbf{k}j$ which undergo a diffuse scattering with probability $1-p_{\mathbf{k}j}$. As a consequence $N^{>}_{\mathbf{k}j} = N^{<}_{\mathbf{k}'j'}$, the number of particles in specularly connected modes is conserved. This strategy allows to completely define the scattering in term of the specularity parameter $p_{\mathbf{k}j}$. We obtain finally

$$P^D(\mathbf{k}j\to\mathbf{k}'j') = (1-p_{\mathbf{k}j})\frac{(1-p_{\mathbf{k}j})\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n}\theta(-\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n})}{\sum_{\mathbf{k}'j'}(1-p_{\mathbf{k}j})\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n}\theta(-\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n})}. \tag{33}$$

The scattering model given by Eqs. 24, 26 and 29, is easily translated algorithmically, and implemented in the following way: at first a random number r is drawn from a uniform distribution in $[0,1[$. If $r < p_{\mathbf{k}j}$ the scattering is specular and $\mathbf{k}'j'$ is computed from Eqs. 22 and 23. Otherwise, the scattering is diffuse and $\mathbf{k}'j'$ is drawn from the probability

$$P = \frac{(1-p_{\mathbf{k}j})\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n}\theta(-\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n})}{\sum_{\mathbf{k}'j'}(1-p_{\mathbf{k}j})\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n}\theta(-\mathbf{v}_{\mathbf{k}'j'}\cdot\mathbf{n})}. \tag{34}$$

The population of this newly determined mode is obtained from the Bose-Einstein distribution at the local temperature, $n^0_{\mathbf{k}'j'}(T)$.

For a particle at position $\mathbf{r}$, the position of its collision, $\mathbf{r}_c$, on a surface is obtained using that $\mathbf{r}_c - \mathbf{r} \propto \mathbf{v}$ and $\mathbf{n}\cdot(\mathbf{r}_c - \mathbf{r}_o) = 0$, where $\mathbf{v}$ is the velocity of the particle and $\mathbf{r}_o$ is any point of the surface used as its plane origin. This gives $\mathbf{r}_c = \mathbf{r} + [(\mathbf{r}_o - \mathbf{r})\cdot\mathbf{n}]/[\mathbf{v}\cdot\mathbf{n}]\mathbf{v}$. From this, the number of timesteps N_t of duration Δt until the collision can be calculated as $N_t = \frac{\|\mathbf{r}_c - \mathbf{r}\|}{\|\mathbf{v}\|\Delta t}$. The collision position $\mathbf{r}_c$ for each particle is calculated for every facet and, for the facets where $N_t \geq 0$, converted to the barycentric coordinates for the triangles that compose that facet. The $\mathbf{r}_c$ that is kept as the true one is that

with minimum positive N_t that has all barycentric coordinates between 0 and 1 for a triangle. Figure 6 shows an illustration of the collision detection for a single particle. This allows to complete the procedure for boundary scattering described as follows:

1. Detect true $\mathbf{r}_c$ and N_t . Update N_t every iteration by subtracting 1;
2. When $N_t \leq 0$, rewind the particle to the collision position $\mathbf{r}_c$, and compute the time $\tilde{t}$ left to the particle to complete the time step;
3. Draw a random number and compare it with $p_{\mathbf{k}j}$;
4. Assign a new mode $\mathbf{k}'j'$ to the particle for specular or diffuse scattering;
5. Calculate the drift for the remaining time $\tilde{t}$ in the timestep (Eq. 11);
6. If the particle collides again, repeat the procedure.

The iteration is concluded and a new iteration starts. To sum up the whole process, particles are added or removed by the reservoirs, drift through the domain (Eq. 11), energy density is updated (Eq. 8), phonon-phonon scattering and boundary reflections occur. Over time the system reaches steady state, and the temperature distribution, heat flux, and thermal conductivity of the nano-component can be analysed.

2.4.6. Thermal conductivity

The thermal conductivity κ can be estimated globally for the total geometry, or locally for each of its subvolumes, by applying Fourier's law in the direction of the temperature gradient of interest.

The global thermal conductivity can be estimated for sliced geometries only, since they have a defined temperature gradient, and hence a direction in which κ can be analysed. The total temperature gradient is imposed, considering each reservoir as one extra slice, and the global heat flux is calculated considering all particles in the geometry,

$$\mathbf{\Phi}_{global} = \frac{1}{N_p/(3N_aN)} \frac{1}{V_0 N} \sum_{i=1}^{N_p} \hbar\omega_i \delta n_i \mathbf{v}_i \tag{35}$$

where N_p is the instantaneous number of particles in the domain. This gives

$$\kappa_{global} = -\frac{\mathbf{r}_2 - \mathbf{r}_1}{T_2 - T_1} \cdot \mathbf{\Phi}_{global} \tag{36}$$

where $\mathbf{r}_s$ and T_s are the position and temperature of the reservoir number s.

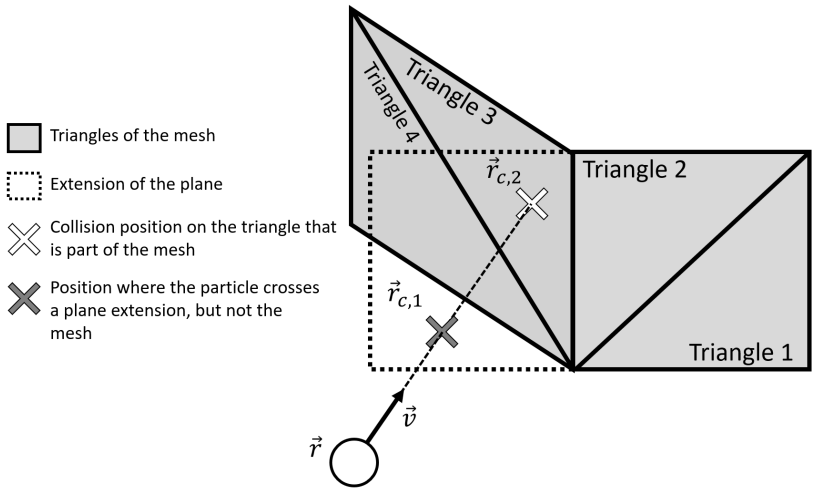

Figure 6: Exemplification of the collision detection process considering two crossing facets. The algorithm calculates the intersection of the path on both planes, $\mathbf{r}_{c,1}$ and $\mathbf{r}_{c,2}$. The former is located out of bounds of triangles 1 and 2, and is therefore discarded. The latter is located within bounds of triangles 3 and 4, and has valid barycentric coordinates for triangle 3. This validates the collision on that facet.

However, since N_{ps} is the number of particle per mode in subvolume s, we have

$$\frac{N_p}{3N_aN} = \sum_s N_{ps}. \tag{37}$$

This gives

$$\begin{aligned}
\mathbf{\Phi}_{global} &= \frac{1}{\sum_s N_{ps}} \frac{1}{V_0 N} \sum_{i=1}^{N_p} \hbar\omega_i \delta n_i \mathbf{v}_i && (38)\\
&= \sum_s \frac{V_s}{V} \Big(\frac{1}{N_{ps}} \frac{1}{V_0 N} \sum_{i\in s} \hbar\omega_i \delta n_i \mathbf{v}_i \Big) && (39)\\
&= \sum_s \frac{V_s}{V} \mathbf{\Phi}_s && (40)
\end{aligned}$$

where we have defined the local heat flux in the subvolume s,

$$\mathbf{\Phi}_s = \frac{1}{N_{ps}} \frac{1}{V_0 N} \sum_{i=1}^{3NN_aN_{ps}} \hbar\omega_i \delta n_i \mathbf{v}_i. \tag{41}$$

A local thermal conductivity can then be calculated in two ways. If the geometry is sliced, then κ_s, the thermal conductivity in subvolume s can be estimated by considering the local heat flux in the subvolume s and estimating the local temperature gradient from finite differences between the adjacent subvolumes, denoted as $s-1$ and $s+1$. This gives

$$\kappa_s = -\frac{\mathbf{r}_{s+1} - \mathbf{r}_{s-1}}{T_{s+1} - T_{s-1}} \cdot \mathbf{\Phi_s}. \tag{42}$$

However, if the subvolumes are generated from a grid or Voronoi polyhedra, then a local thermal conductivity is calculated in between each two subvolumes, denoted as r and s, by taking the average heat flux and the temperature gradient between them. This gives

$$\kappa_{rs} = -\frac{\mathbf{r}_s - \mathbf{r}_r}{T_s - T_r} \cdot \frac{\mathbf{\Phi}_r + \mathbf{\Phi}_s}{2}. \tag{43}$$

and we say that κ_{rs} is the thermal conductivity along the connection between r and s.

2.4.7. Errors, convergence detection and simulation ending

To check the convergence of a calculation, we can monitor several quantities,

- The temperature in each subvolume,
- The heat flux in each subvolume in x, y and z directions,
- The thermal conductivity in each subvolume or along a connection,
- The energy balance in each reservoir.

Every 100 iterations the errors ϵ_i are calculated for each quantity i. A number of data-points defined by the parameter `--n_mean` is used in the calculation, each data-point being 10 timesteps apart from the previous. The mean μ_i and standard deviation σ_i over time are taken. These values are the ones shown in profile plots and in the value of global thermal conductivity. The individual error ϵ_i is calculated as the relative change from the previous mean value:

$$\epsilon_i = \left| \frac{\mu_i^{k+1}}{\mu_i^k} - 1 \right| \tag{44}$$

To avoid the errors shooting up to infinity, any quantity that contains 0 within the interval $[\mu_i - \sigma_i, \mu_i + \sigma_i]$ is considered to have $\epsilon_i = 0$. The maximum individual error of any quantity is considered as the global error ϵ.

It is the user who defines how low ϵ should be, by setting the parameter `--conv_crit`. Two values shall be passed: the first being the desired ϵ and the second value being the number of consecutive checks that shall stay under the desired maximum. If this is achieved, the simulation is considered to have converged. The stochastic nature of Monte Carlo does not allow errors as low as deterministic methods unless a very large number of particles are used, which is usually impractical. Usual ϵ values are in the order of 1%, which allows for fair uncertainty and reasonable execution time.

The simulation can also be interrupted by number of iterations (`--iterations`) or by time limit (`--max_sim_time`). Whenever one of these conditions is reached first, the simulation will be interrupted and results will be post-processed.

2.5. Code structure

2.5.1. Organization

The code organises the relevant information into six classes for better compartmentalisation: Constants, Mesh, Geometry, Phonon, Population and Visualisation. Figure 7 shows the created objects and their relations inside the code, and Fig. 8 shows a flowchart of the algorithm.

- The "Constants class" store all relevant physical constants and conversion factors to be inherited by the requiring classes;
- The "Mesh class" stores the information about a single three-dimensional triangular mesh.
- The "Geometry class" loads, processes and manages all information about the relationships among the triangular mesh, the boundary conditions and subvolumes.
- The "Phonon class" loads and processes material data from the *ab-initio* simulations and stores the relevant methods to calculate energy density, relaxation times and interpolate temperature. Each Phonon object represents a different material.

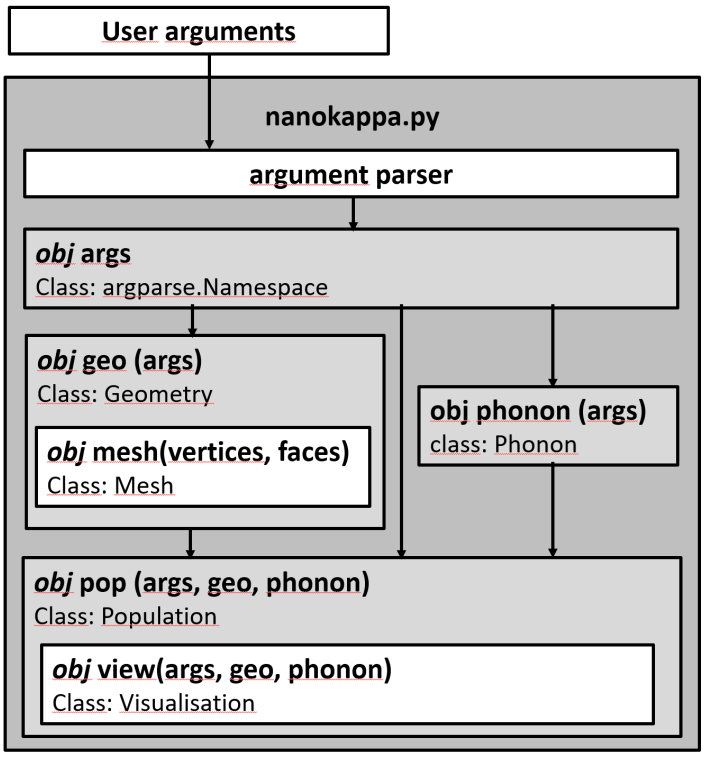

Figure 7: Code structure with relationships between objects.

- The "Population" class is where the simulation itself takes place, using Phonon and Geometry objects as inputs along with additional parameters. It needs therefore one single Geometry object and one Phonon object.

- The "Visualisation class" takes the information from the convergence files and population object and translates it into figures to be analysed by the user as the simulation runs. A Visualisation object is created inside the Population object, and the geometry and phonon objects are passed on by Population to Visualisation.

The main script, `nanokappa.py` takes the input parameters defined on command line or in an external text file and parses them with the "argparse" standard Python module. It then generates objects of the Geometry, Phonon and Population classes, and iterates the calculation until one of the termination conditions has been met.

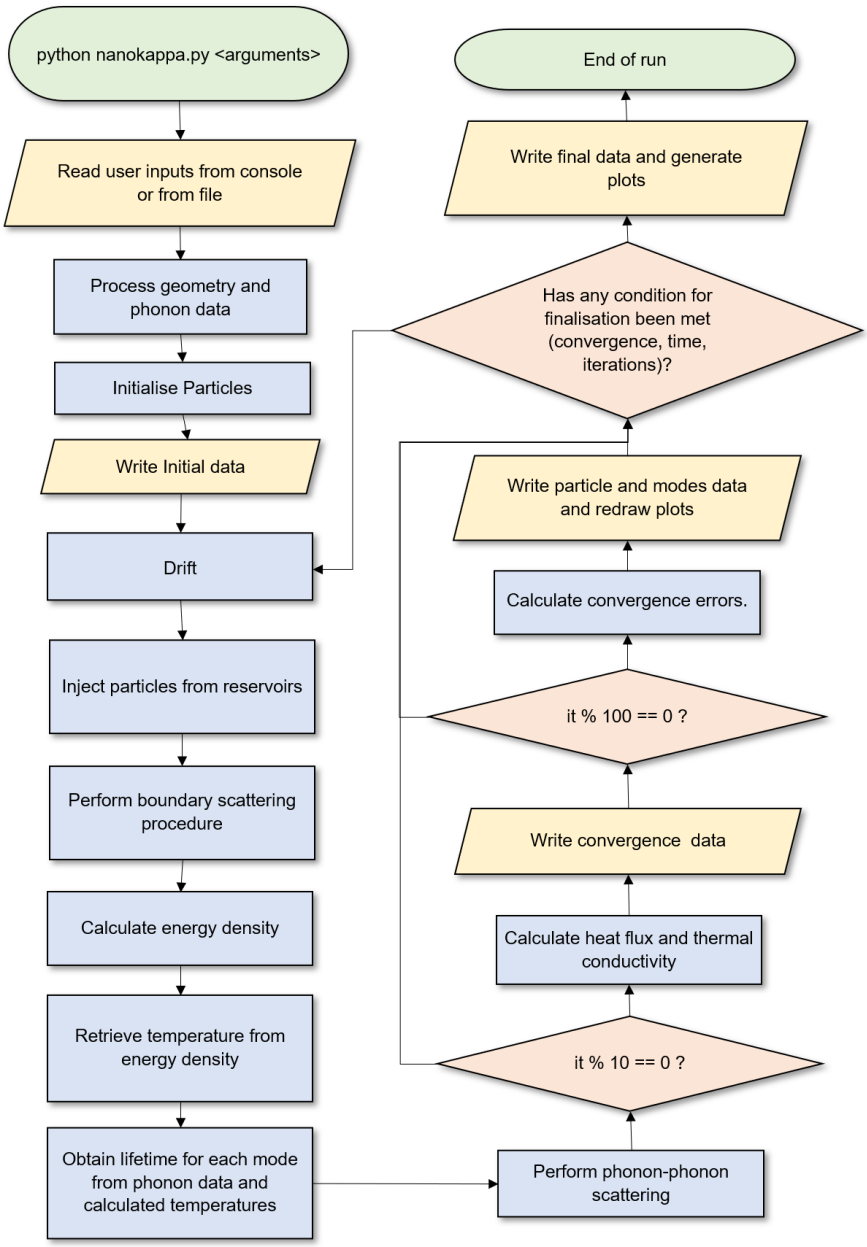

Figure 8: Nano-κ algorithm flowchart.

Particle properties are stored sequentially as "Numpy" arrays, so the code is able to apply vectorised calculations over them. For boundary scattering, operations are applied on a slice of the arrays corresponding to the concerned particles, and the new values are then substituted to the main arrays. Typically, convergence data is stored every 10 iterations. Other data and plots are updated every 100 iterations.

3. Results and discussions

3.1. Sensitivity studies

In the following, a set of simple cases was run to study the sensitivity of the algorithm by varying number of particles, the number of subvolumes and the timestep value, from a reference calculation. Those directly affect the uncertainty and speed of the calculation. Indeed, the more particles a simulation uses, the more precise is the result and the longer it takes to run. This remains a user decision to balance between expected accuracy and simulation duration.

3.1.1. Reference case

The chosen reference case was set as a 2 µm thick silicon thin film, submitted to a temperature gradient $\Delta T = 4\,\mathrm{K}$, where $T_{hot} = 302\,\mathrm{K}$ and $T_{cold} = 298\,\mathrm{K}$. The domain was sliced into 20 subvolumes and populated with 10^6 particles in total. The phonon properties computed from DFT simulations were considered in a Brillouin zone discretised into $31 \times 31 \times 31$ wavevectors. Considering all 6 branches, acoustic and optical, this totalises 178,746 modes. The timestep was set to 1 ps, and the number of timesteps was 10,000, simulating the system for 10 ns. The initial temperature profile was set as varying linearly from T_{hot} to T_{cold}, and the local temperature was used as reference for better statistics. The temperature for each particle was estimated by linearly interpolating between subvolumes. No time limit or maximum error criterion was imposed. Calculation ends after the $10{,}000^{th}$ timestep.

Figure 9 shows the domain used. It was built as a box with 20.000 Å sides, applying the temperature gradient in the x direction. Despite it being possible to use shorter lengths in y and z directions to simulate a thin film, a larger domain decreases the number of boundary scattering calculations for the same end result, which decreases simulation time.

Figure 10 shows the number of particles in the domain, while Figs. 11 and 12 show respectively the temperature and heat flux evolution in each

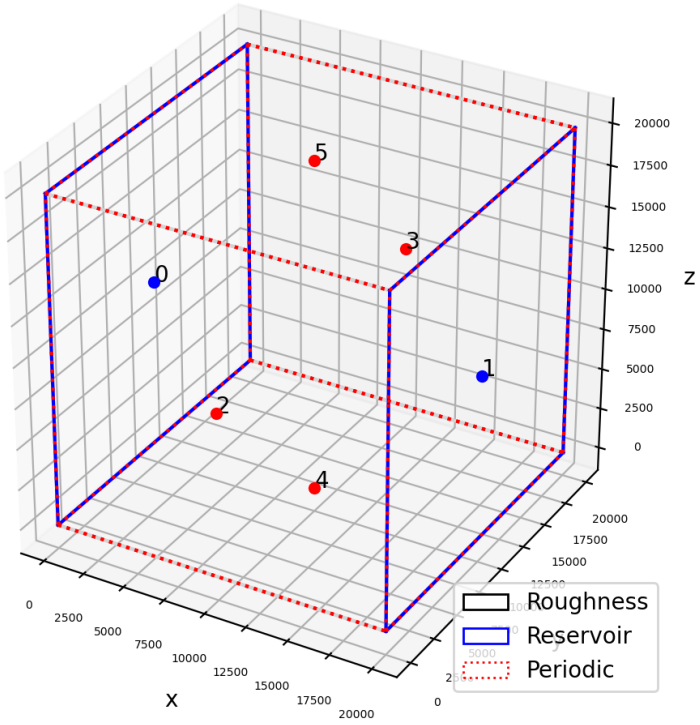

Figure 9: Domain and boundary conditions for the sensitivity study. The points are the "centroids" of each facet. Blue facets show where are the reservoirs with imposed temperature. The red facets are the ones with periodic boundary condition, with facet 2 connected to facet 3, and facet 4 to facet 5. The units of the axis are in Angstroms.

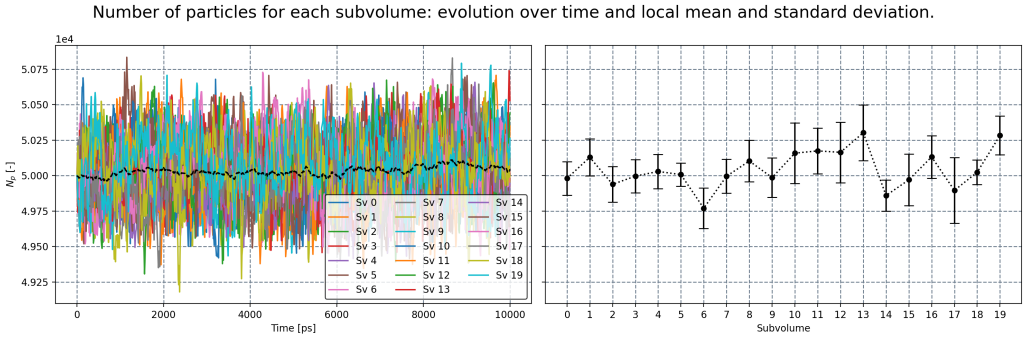

Figure 10: Evolution of the number of particles in the reference case. Left: number of particles in each subvolume over time. Right: Final profile with mean and standard deviations.

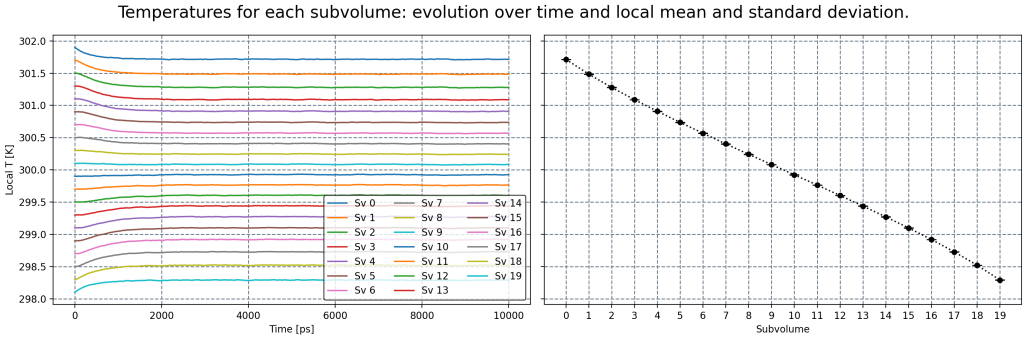

Figure 11: Evolution of the temperature in the reference case. Left: number of particles in each subvolume over time. Right: Final profile with mean and standard deviations.

subvolume. The mean values and standard deviations were computed over the last 20 convergence data-points. In this reference case, initially the 10^6 particles are equally shared into the 20 domain cells. During the simulation, these particles move from a cell to another and may exit or enter through hot and cold black bodies. It clearly appears that this processes does not induce instabilities as the average number of particles in a slice remains close to 5.10^4. Here, the maximum variation of the number of particles in each subvolume is around 1.5%, from the expected 5.10^4, and a maximum variation in the whole domain is less than 0.5% from the expected 1 million.

Considering Fig. 11, the temperature profile in the right plot slightly bends away from the linear profile predicted by the Fourier regime at macroscale. This curvature is due to the ballistic transport of the phonons coming from

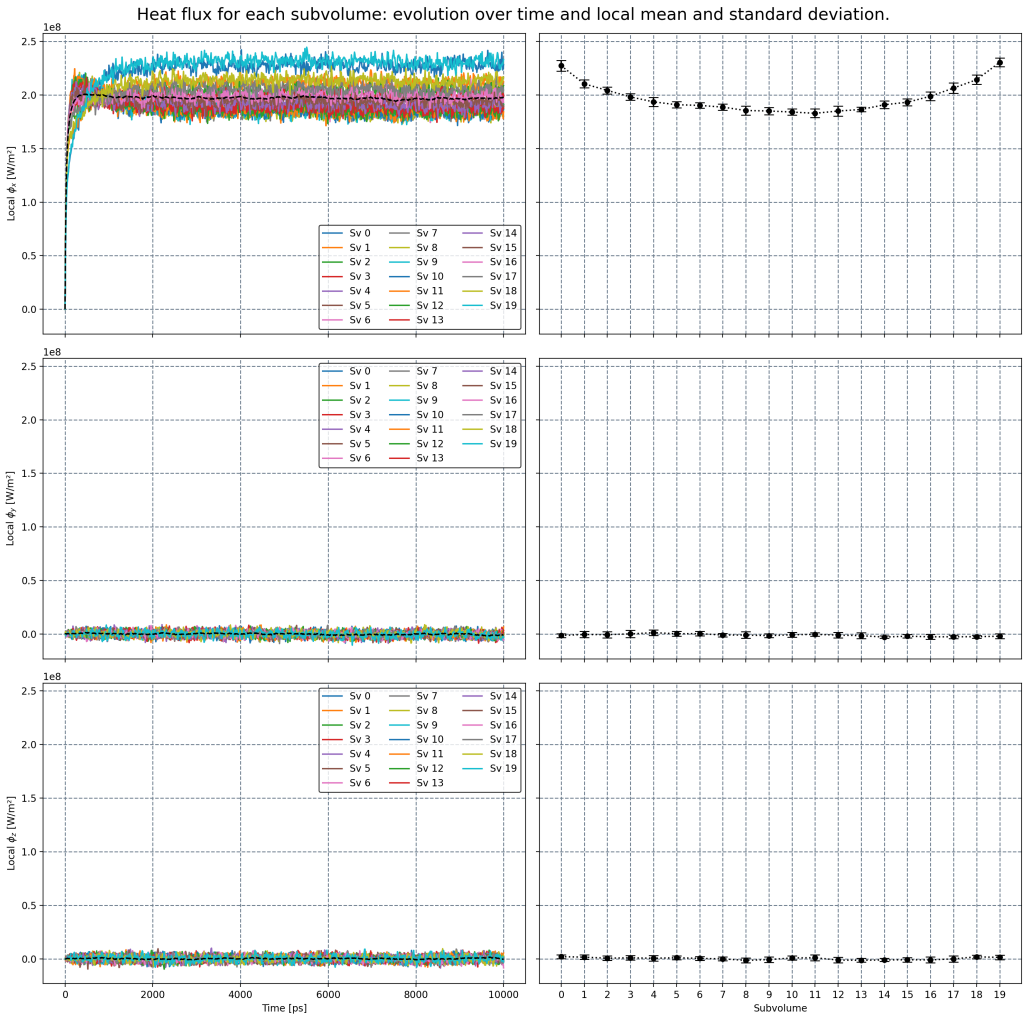

Figure 12: Evolution of the heat flux in x, y and z directions in the reference case. Left: number of particles in each subvolume over time. Right: Final profile with mean and standard deviations.

the reservoirs. The deviation can also be seen on the left plot, where the initial temperature is the linear profile expected by Fourier's law. This effect is more pronounced at smaller length scales and lower temperatures. In the first case, the higher temperature gradient allows phonons to keep a high deviation from equilibrium distribution $n^0_{\mathbf{k}j}(T)$ as they travel in the domain, allowing the temperature field to go away from the Fourier regime. In the second case (low temperature) this leveling happens because of longer relaxation times which allow phonons to travel farther without giving off energy. Therefore, the distribution of energy is kept almost constant in the domain, and consequently the temperature profile.

Since the temperature gradient is applied in the x direction, while periodic BC are set in y and z directions, the energy carriers flow along x, and thus in Fig. 12 ϕ_y and ϕ_z are kept around zero throughout the simulation. The heat flux ϕ_x increases with time in the whole domain, starting from an initial null value. As the modes transfer energy among each other and the temperature gradient is adapted, the heat flux profile stabilises in a curved shape. This shape is due to the ballistic transport on the edges of the system. The higher temperature gradient at the borders induces a larger heat flux than in the center.

Figure 13 shows the behaviour of the thermal conductivity during the simulation. In the top left, the linear temperature gradient from the beginning helps to stabilise the values of κ_s quickly after simulation start. The effect of ballistic transport is observed in the thermal conductivity profile in the top right plot (see Fig. Fig. 12). In agreement with what was observed for temperature and heat flux spatial variation, direct evaluation of "local" (i.e. per slice) thermal conductivity is somewhat biased as Fourier's hypothesis are not fulfilled on the edges of the sample. In this part of the spatial domain, heat flux is not constant while thermal gradient remains globally the same. This behaviour vanishes as samples are longer, temperatures higher or in systems with increased boundary scattering. This is consistent with the more "diffuse" behaviour expected in the latter situations. Anyway, the global κ, however, arrives relatively quickly to a steady value, since it is calculated with the fixed boundary temperatures and depends only in the global heat flux. It means therefore that the final global flux of energy is rapidly approximated, and is only adjusted as the local temperatures and phonon distributions arrive to a steady state.

Another important quantity to keep track is the balance of energy in the system considering wave-packets entering and leaving at blackbody BCs.

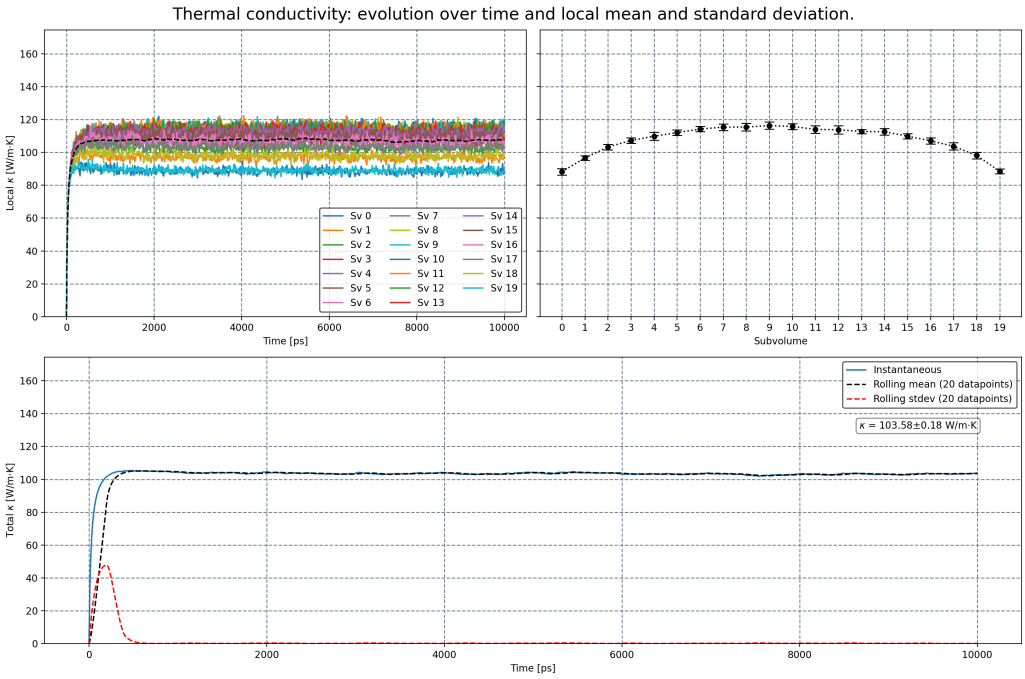

Figure 13: Convergence of thermal conductivity for each subvolume (top) and globally (bottom).

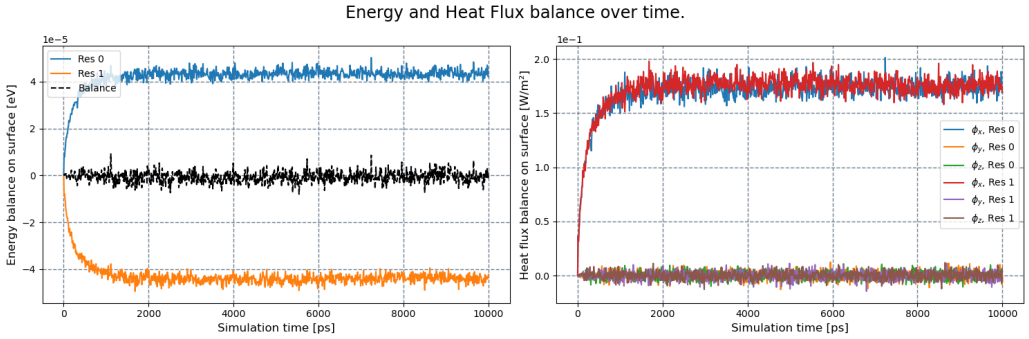

Figure 14: Convergence of energy balance and heat flux balance in the domain.

Hence, Fig. 14 (left) shows the energy balance (net sum of carrier's energy entering/leaving) on each facet in contact with the hot side (blue curve) and the cold side (orange curve) reservoirs, while on Fig. 14 (right) is plotted in which direction this energy is crossing these boundaries considering heat fluxes. As expected, the global energy balance curve (in black) is kept around 0 eV, since the net energy entering from one reservoir and leaving to the other is approximately the same for the symmetric configuration. In the present case, it takes roughly 4000 ps for the system to have achieved steady state. On the perspective of heat flux, the values of net heat fluxes referring to both hot and cold reservoirs are positive, since their directions are the same $(+x)$.

3.1.2. Number of subvolumes

The first sensitivity study was done by changing the number of subvolumes while keeping the number of particles constant at 10^6. Because the number of subvolumes affects the smoothness of ∇T, each number of subvolumes was simulated with "linear" and "nearest" value interpolation. In the "linear" case, temperature profile in the system linearly varies from the center of a cell to the center of the following cell, allowing to perform Bose-Einstein and lifetime calculations, that both depends on T, at a temperature close to the expected one. In the "nearest" case, the temperature profile in the system is "step-like", keeping constant T value on the whole cell. Obviously, increasing the number of subvolumes makes the difference between linear and steps profiles vanishing. The results of system thermal conductivity are shown in Fig. 15. The calculated system κ starts very high, it decreases as the number of subvolumes is increased and stabilizes around $100\,\mathrm{W/m \cdot K}$. This value agrees with the literature, as it will be shown in the next section (Sec.

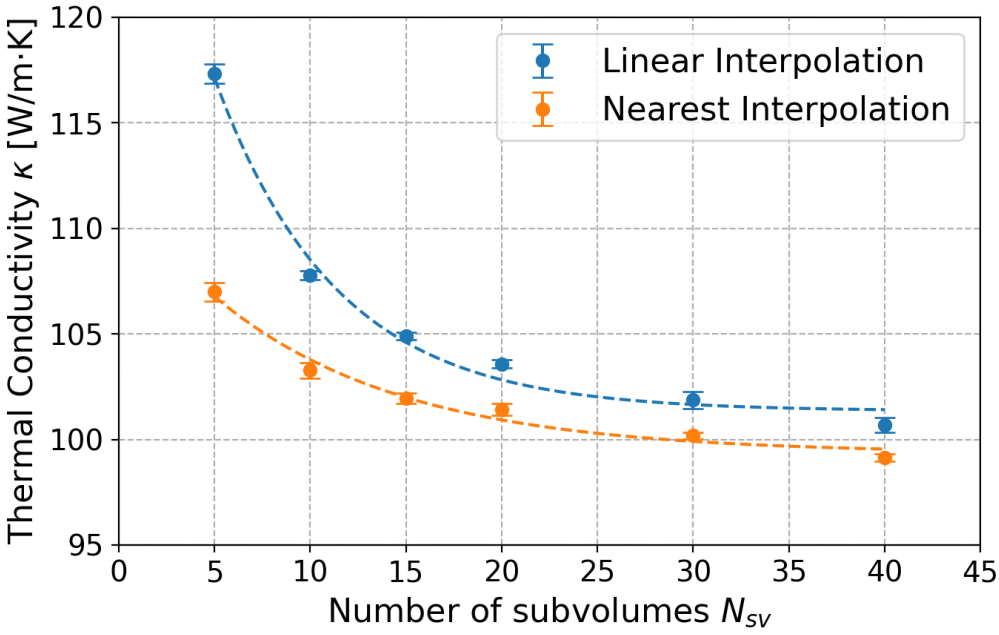

Figure 15: Variation of thermal conductivity as function of number of subvolumes.

3.3). The κ overestimation occurs because, with longer slices, the estimation of temperature is calculated in subvolumes that are too large as compared to the local equilibrium assumption. This phenomena mostly affects the system borders where ∇T is steeper. The difference between "linear" and "nearest value" interpolation is due to what each particle "locally sees" when scattering. In the "nearest value" interpolation, the temperature is kept constant inside a subvolume, and particles with short mean free path can easily reach equilibrium state. This does not happen in the "linear" interpolation, since the particle is always kept at a distance from the $n^0(T)$ every-time it drifts, even when it does not change subvolumes. This increases the obtained value of $\mathbf{\Phi}$ and, naturally, κ. However, as the number of subvolumes increase, the two methods get closer to each other. The uncertainties remain stable around $0.2\,\mathrm{W/m \cdot K}$, indicating that the number of subvolumes affects more the mean value than the standard deviation of the result.

3.1.3. Number of particles

Figure 16 shows the results of the sensitivity study with respect to the number of particles used in the simulation. Two groups of simulations were run: one with local T_{ref}, that change from cell to cell, and another with $T_{ref} = 300\,\mathrm{K}$. There is no noticeable effect regarding the mean κ value calculated using each method. The calculated κ is found to be stable for all simulations, around $103\,\mathrm{W/mK}$.

The major effect can be seen in the standard deviation variation. The more particles are simulated, the more precise is the estimation of κ, both locally in each subvolume and globally in the entire domain. For large N_p, the time spent in each iteration increases proportionally, being at the end for the user to decide the acceptable σ_κ. Regardless, the calculations with local T_{ref} has consistently less noise than the simulations with fixed reference, with an approximate σ_κ being obtained with 10 to 15 times less particles (note the log x axis).

3.1.4. Timestep duration

The timestep duration was also varied, with the total simulated time being kept the same, at 10 ns. Both time discretisation schemes (linear and exponential) exposed in Section 2.4.4 were compared. The results are shown in Fig. 17. In this case, the number of data-points used for the average and standard deviations evaluations must be changed, so that they are calculated over the same time interval. By doing this, with the exponential scheme, we

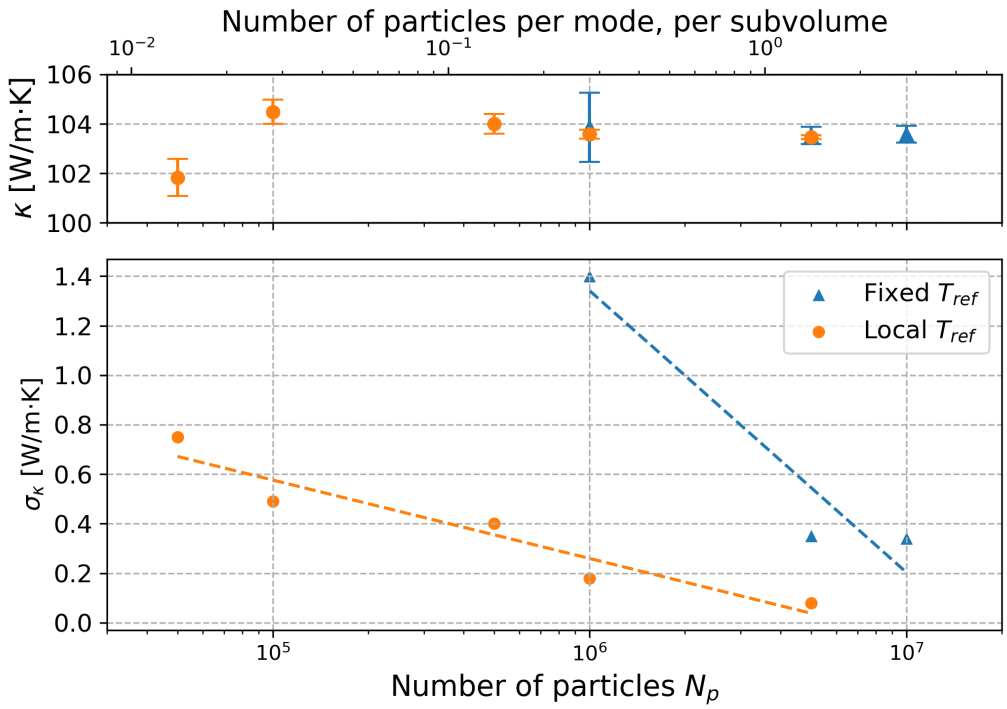

Figure 16: Variation of thermal conductivity as function of number of particles. In the detail, the variation of the uncertainty with N_p.

can see an upward trend as timestep size increases. This increase is caused by the longer travelled distance of the phonons in a single timestep, carrying their energy farther without being affected by scattering, which in turn increases $\mathbf{\Phi}$ and consequently κ. A more correct estimation of interaction between phonons is achieved by smaller timesteps. Here, below 1 picosecond, thermal conductivity does not change much. Yet, the simulation time needed to achieve steady state is inversely proportional to Δt which can be an issue in the case of long systems. To tackle this issue, further numerical developments such as parallelisation are needed.

The results with the linear scheme are also shown in Fig. 17. In the latter case, the largest timestep that could be applied was 2.5 ps. Above, the simulations crashed due to the instabilities shown in Section 2.4.4. This failure in calculations are due to the modes with shorter lifetimes that are the ones that become unstable first, even if they weakly contribute to the overall thermal conductivity. On the other hand, the modes with longer relaxation times are not affected as much by the application of the linear scheme, and their contribution to κ is kept stable. To conclude on this issue, for materials with short lifetimes, the maximum dt needs to be reduced accordingly to

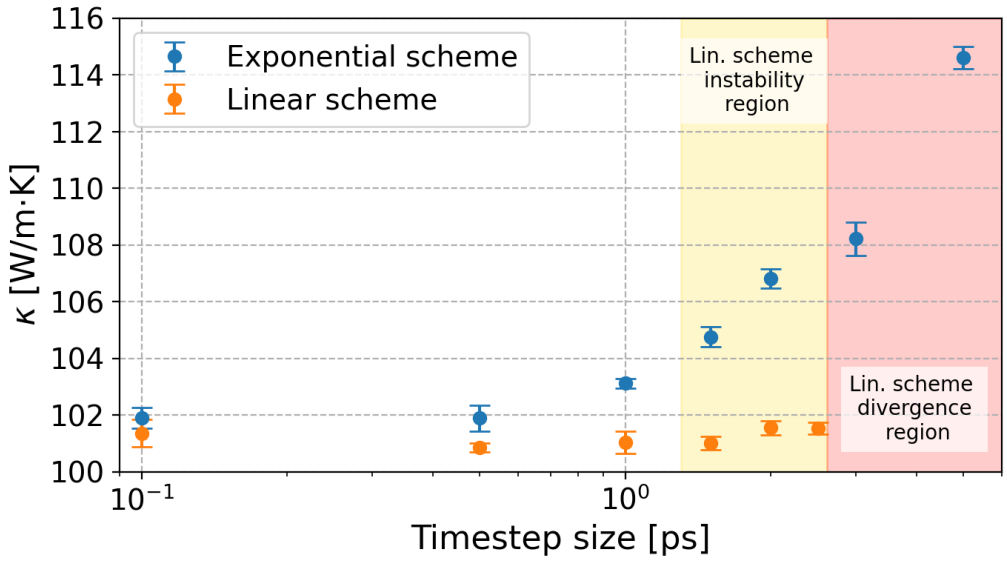

Figure 17: Variation of thermal conductivity as function of timestep for each time discretisation scheme. The instability regions for the linear scheme are highlighted.

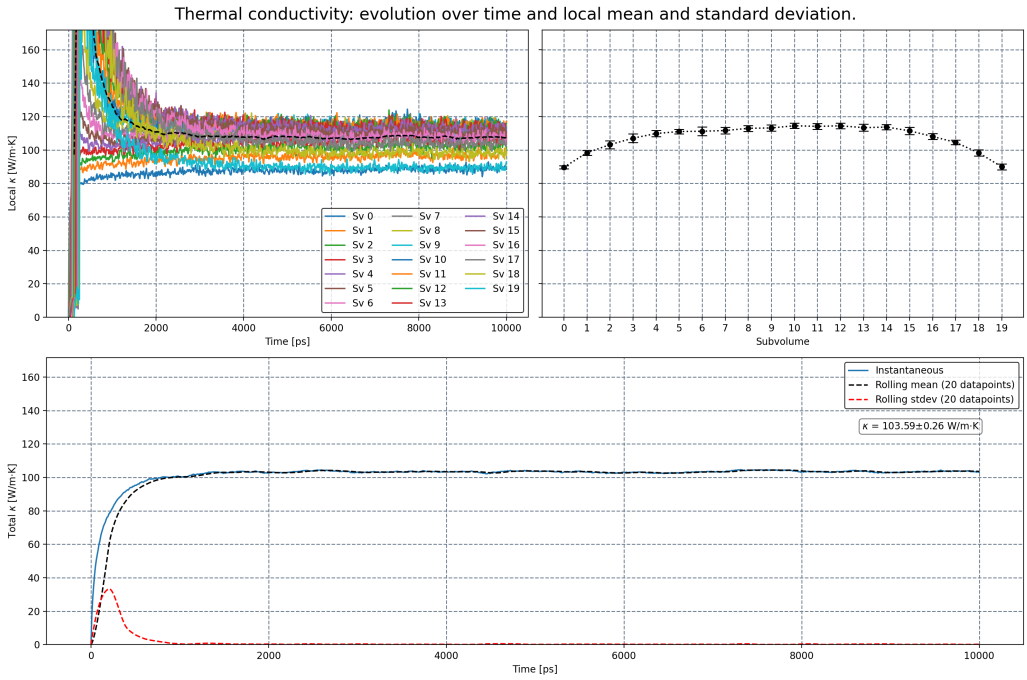

Figure 18: Convergence of local and global thermal conductivity with a constant cold initial T profile.

ensure convergence and reliability.

3.1.5. Initial conditions

It is also of interest to see whether the initial conditions have any influence on the final quantities. The main initial condition is the initial temperature distribution. For example, the simulation can be started with a linear temperature variation between hot and cold baths, or the whole domain can be set at T_{cold} as it is done hereafter. Figure 18 shows the convergence of κ_{sv} and κ_{global} for an initially constant profile as such. In the simulation with initially constant profile, the temperatures of neighboring subvolumes are very close at the beginning and, thus, the thermal gradient is small. As a consequence, as shown in the top left plot, any small variation of ∇T induces strong fluctuation of the local thermal conductivity until the body is sufficiently heated and energy starts to find its way towards the colder side of the thin film. Also for this reason, the last subvolume's thermal conductivity stabilises much later than the first ones. This differs from the behaviour

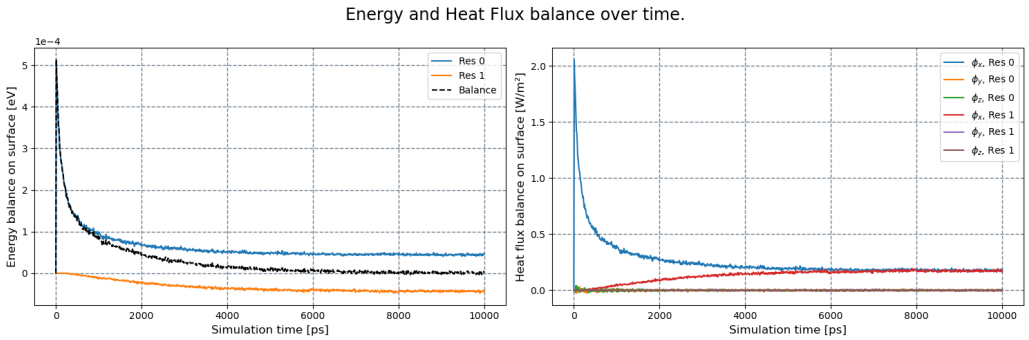

Figure 19: Convergence of energy balance with a constant cold initial T profile.

of the base case, where all thermal conductivities rise together as the heat flux increases. Comparing the plots at the bottom of each figure (Figs. 13 and 18), it can be seen also that the global κ approaches more quickly its final value with a linear initial temperature profile condition. Since κ_{global} is calculated based on fixed ΔT, it depends only on $\mathbf{\Phi}$, which in turn is much faster adjusted because of the previously established temperature gradient.

The delayed convergence can be observed as well in the energy and heat flux net balance. Figure 19 (left) shows the energy balance for the constant cold case. In contrast with the data shown in Fig. 14, the balance of energy starts high, as the only net energy exchange with the reservoir happens at the heated side. As time passes and high energy phonons start to reach the opposite side, the balance approaches zero and stabilises, but only near the 10 ns limit. In other words, based on the convergence of κ and energy balance alone, the initial linear profile can offer convergence, in a limited number of timesteps, that is at least half of that needed by the initial cold profile.

Another initial condition that can be set is the constant mean temperature, in this case 300 K. The behaviour is similar to the constant cold T case, but the convergence is as faster as the linear case. This happens because the system is already heated with an overall energy close to the final one, being only necessary to adjust the phonon distribution to the heat flux.

3.1.6. Final observations

Given these preliminary observations, the following study cases were ran with the following specifications:

- The number of subvolumes was varied by keeping each subvolume 500 Å long;

- The density of particles per length of film or wire was kept the same by fixing 10^6 particles for every 2 µm;
- The reference temperature was always set as local;
- The initial temperature profile was set as linear;
- The timestep was 1 ps;
- The average values were taking with 100 timesteps, or 100 ps.

Dimensional parameters such as film thickness or wire diameter were changed according to the available experimental data.

3.2. Thin films - cross-plane conduction

Several cases of silicon and germanium thin films were simulated to show the effects of temperature and thickness in thermal conductivity. Figure 20 shows the results for both analyses.

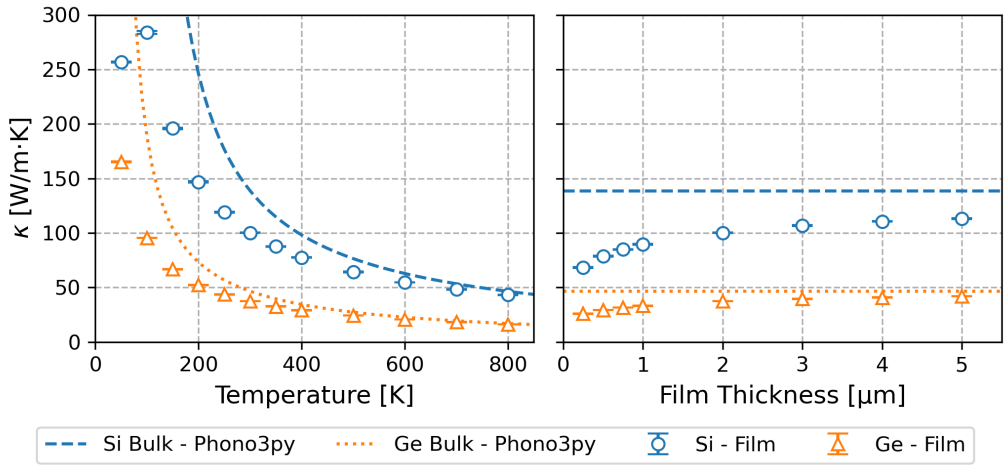

Figure 20: Thermal conductivity of 2 µm thin film as a function of temperature (left); Thermal conductivity at 300 K of thin film as a function of film thickness variation (right), for both Si and Ge.

In figure 20 (left) the expected thermal conductivity variations are recovered for both Si and Ge. In the present case, 2 µm thin film, thermal conductivities remain smaller than in the bulk counterparts for temperature bellow 500 K. Above, our simulation results are in good agreement with

the literature. This is consistent with the fact that phonon transport is dominated by phonon-phonon scattering and energy carriers' mean free path becomes small. Bellow 500 K size effects matter as the distribution of phonon mean free path can be larger than the system size. For the coldest temperatures, the relaxation time of phonons is greatly increased. If the distance between thermal reservoirs is short enough, there will be a point where the ballistic transfer between reservoirs will become more important and thermal conductivity will decrease. This is visible below 100 K for the Si film. For germanium, however, this limit is below the studied interval as Ge phonon lifetime is globally smaller than for Si.

In figure 20 (right) the thermal conductivity is plotted versus film thickness at room temperature. As the thickness increases, the closer to the Fourier regime the heat transfer is. With less ballistic effects, κ gradually approaches its bulk value ($138.7\,\mathrm{W/m\cdot K}$ for Si and $46.6\,\mathrm{W/m\cdot K}$ for Ge, extracted from DFT-based calculations with our dispersion and lifetimes).

In addition to this general information, one interesting aspect to observe is the contribution of each frequency to the total heat flux. Figure 21 shows the sum of $\hbar\omega n\vec{v}$ for each frequency interval. Presently the full phonon spectra of silicon ($\omega_{max} = 94.6\,\mathrm{THz\,rad}$) is discretised over 100 frequency intervals with same width. The curve is smoothed out to facilitate analysis. In the latter figure, each column correspond to a different film thickness (500, 1000 and 2000 nm). Top row shows the distribution for the first subvolume, near the hot reservoir, while the bottom row shows the distribution for one subvolume near the middle of the film. In those calculations, the number of subvolumes (10, 20 and 40 respectively) is adjusted to keep same slice length.

First, it can be noticed a reduction of the heat flux peaks, for all polarization branches, from left to right, as the system length increases. This reflects the fact that keeping the same hot and cold BCs the amount of energy flowing in the system naturally decreases with a increase of length. Second, we can see a major contribution of the three acoustic mode polarisations (Branches 0, 1 and 2 on sub-figures) as compared to the optical branches depicted by the second peak above $60\,\mathrm{THz\cdot rad}$. However contribution of the latter is not negligible and represents nearly 14% of the cumulated heat flux in the first subvolume, and 8 to 11% in the middle section of the film. Third, comparing the top and bottom rows, it can also be seen how the phonon-phonon interactions change the shape of the distribution along the same film as energy gets redistributed among modes. In particular we can observe a decrease of large frequency acoustic mode peaks (between 30 and $40\,\mathrm{THz\cdot rad}$) toward

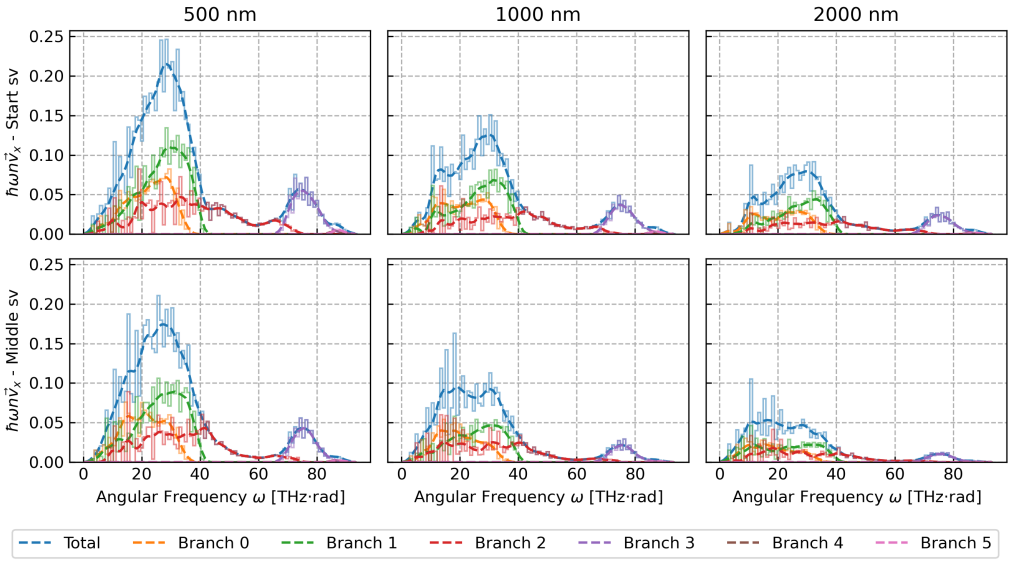

Figure 21: Contribution of frequency in the total heat flux for three different film thickness.

low frequencies, corresponding to the scattering of phonons close to the edge of the FBZ (Umklapp processes).

The same analysis can be done for temperature variation. Figure 22 shows the distributions for 100 K, 300 K and 500 K. For low temperatures, longer relaxation times allow higher heat flux and acoustic modes are dominant. As temperature increases, the total flux is reduced and optical modes have more importance in the energy transport in agreement with Bose-Einstein distribution.

The results were also compared with experimental data by Asheghi et al. [1] for temperature variation and Scott et al. [25] for film thickness variation, both for Si. The thermal conductivity of the film was calculated in two ways: first, considering the ΔT between reservoirs, secondly considering an "internal" ΔT related to local temperature variation between subvolumes of the domain. At low temperatures or thin thickness, however, when there is a high ballistic effect on phonon transport, the actual temperature gradient inside the film is drastically reduced, which affects the estimation of κ. For this reason, the conductivity calculated considering the internal temperature gradient was also added to the plot. The results show good agreement with experimental data. For low temperatures and thinner films, as expected, there were higher deviations between both, total and internal, calculated κ

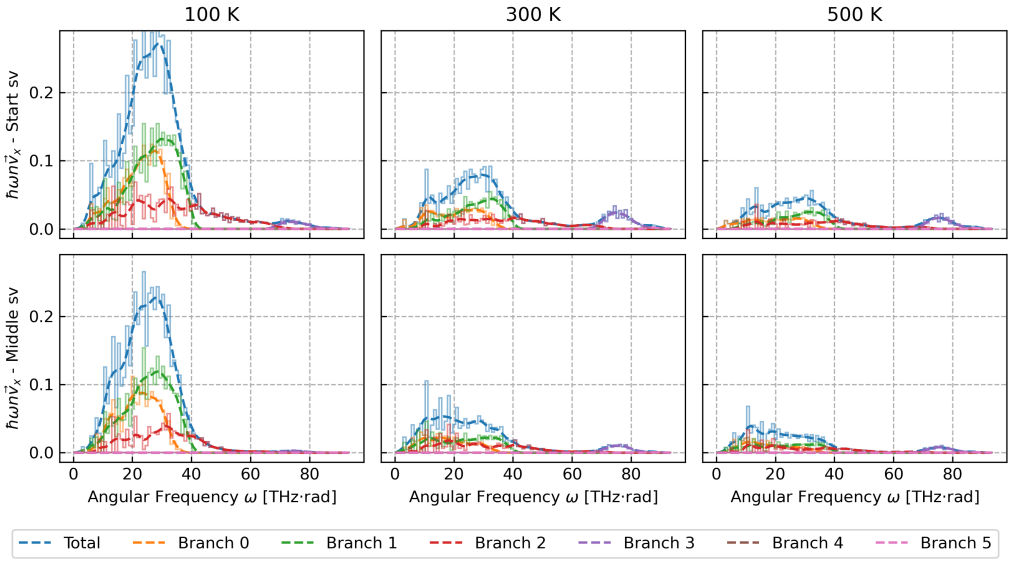

Figure 22: Contribution of frequency in the total heat flux for three different temperatures.

values. Nevertheless, the estimates from simulations still work well as an upper and lower bounds for average values in low temperatures.

3.3. Thin films - in-plane conduction

In order to investigate the in-plane thermal conductivity, the thickness of the film was varied. Concerning boundary conditions, two of the "regarding" facets are set as periodic while the two others are set as rough faces. Figure 25 shows a scheme. The rough facets were considered to have a mean roughness of $\eta = 1\,\text{nm}$. The varying thickness is applied between the rough facets. The results for κ are shown in Fig. 26.

As we have a very thick film and start to thin it, the thermal conductivity shows little variation, with a value close to the cross-plane conductivity (here, thickness along y varies, while it remains constant and equal to $2\,\mu\text{m}$ along x and z directions). As the rough facets are brought closer and closer, more particles scatter on them and are diffusely reflected, decreasing heat flux and therefore thermal conductivity.

Some simulations were also executed to compare results with experimental data from Ju and Goodson [15], Liu and Asheghi [20], Hao et al. [13], Aubain and Bandaru [2, 3], Chávez-Ángel et al. [7], Cuffe et al. [9] and Bosseboeuf et al. [4]. For this, the thickness was varied from $20\,\text{nm}$ to $1\,\mu\text{m}$.

The roughness was set to very high ($\eta = 10\,\mathrm{nm}$) to ensure high diffusivity. Results are shown in Figure 27. The results show good agreement with experimental data in the entire treated range. The variation shown by the experimental data is due to different conditions and methods of measurement, as well as film growth techniques. Nevertheless, the presented trend is the same, with higher conductivity for thicker films. Due to the high roughness set as boundary condition, the κ calculated with the total ΔT is very close to the the one calculated with internal ΔT, acknowledging weak effects related to ballistic phonons.

3.4. Nanowires

Investigating nanowires was a first attempt to increase complexity of the system geometry. For the analysed nanowires, two different shapes were tested. A first set was considered with a square cross-section and a second one had a discretised quasi-circular cross-section (polygonal cross-section with 20 sides). Both shapes were defined in order to have the same cross-sectional

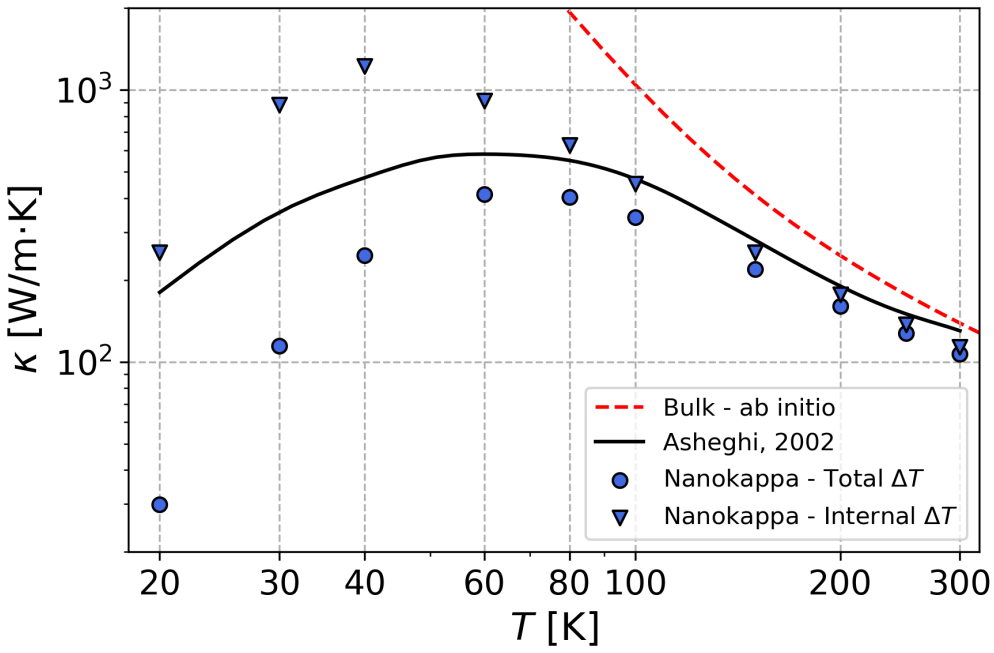

Figure 23: Simulation results for film cross-plane conductivity as function of temperature compared with experimental data referenced by Asheghi et al. [1].

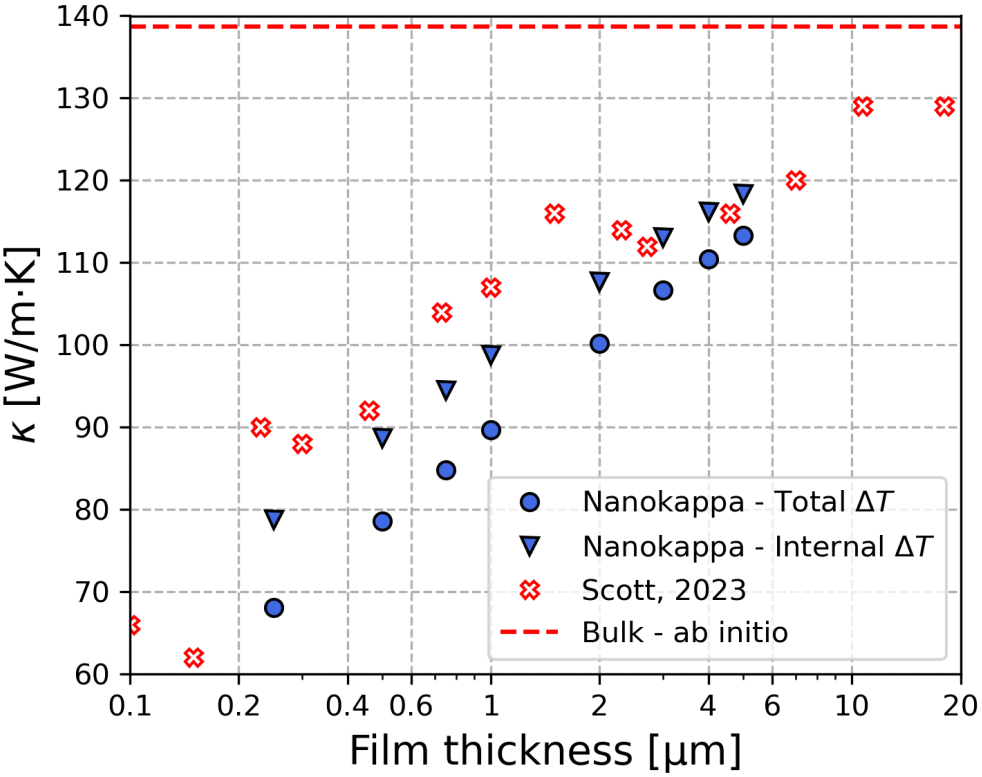

Figure 24: Simulation results for film cross-plane conductivity as function of film thickness compared with experimental data referenced by Scott et al. [25].

area. For both of them, the surface roughness of the walls were varied, and κ was compared. Figure 28 shows the obtained values.

As expected, in the square wire increasing wall roughness reduces the thermal conductivity. Yet, what is more interesting is the fact that the quasi-circular wire already starts with a small thermal conductivity. This is due to the polygonal side orientation that do not allow specular reflections to all modes as the crystal lattice is no longer aligned with those facets like in the square section case. This considerably increases diffuseness behaviour even without any rough walls. The curves for thermal conductivity of both shapes asymptotically approach nearly the same value, around $49\,\mathrm{W/m \cdot K}$. The latter can be considered to be the same after 2 nm roughness. No difference in temperature profile and heat flux was also observed between the two shapes

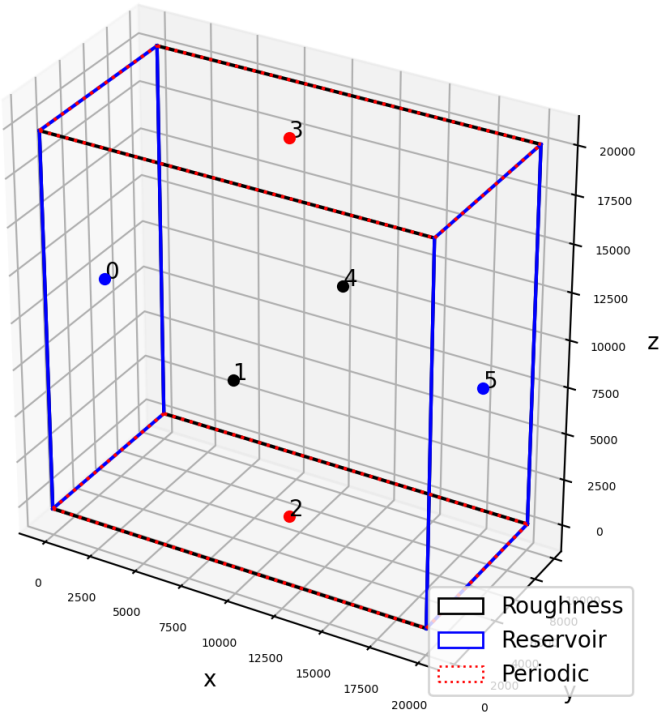

Figure 25: Example of boundary conditions set to evaluate in-plane conductivity. Red facets are considered as periodical, black facets have an associated roughness, blue facets have a set temperature.

when roughness was higher than this threshold.

The comparison between simulation and experimental is shown in Fig. 29. For this the diameter of the nanowire was varied between 22 nm and 300 nm. The simulated wire had a quasi-cylindrical cross section with 20 sides, and high roughness. The plots show good agreement with the experimental data, specially for large diameters. At small diameters, the conduction is slightly overestimated. Similarly to the analysis of thin films, the experimental inputs used for comparison relate to data-points obtained with different methods, wires with slightly different configurations (grown in different directions, poly or monocrystalline, etc.) and sometimes with missing information or error bars. Nevertheless Nano-κ was able to give a good estimate of the thermal conductivity in the entire diameter range.

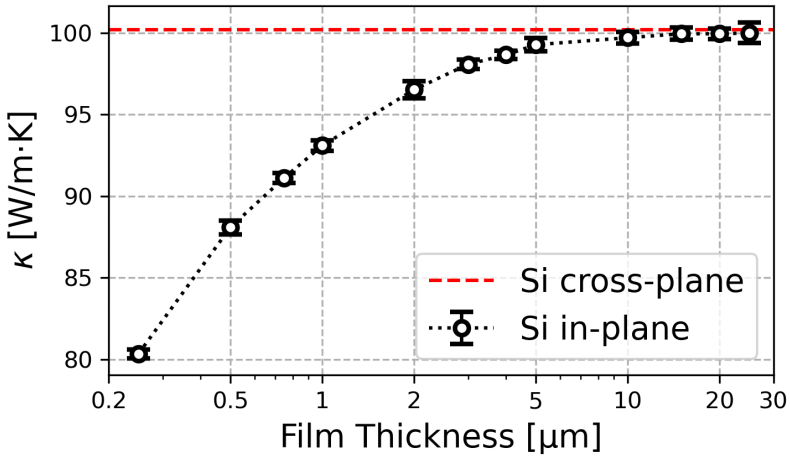

Figure 26: In-plane thermal conductivity of Si thin films as a function of film thickness and comparison with calculated cross-plane result with similar simulation parameters; length between thermostats is 2 µm; thermostats at 302 K and 298 K.

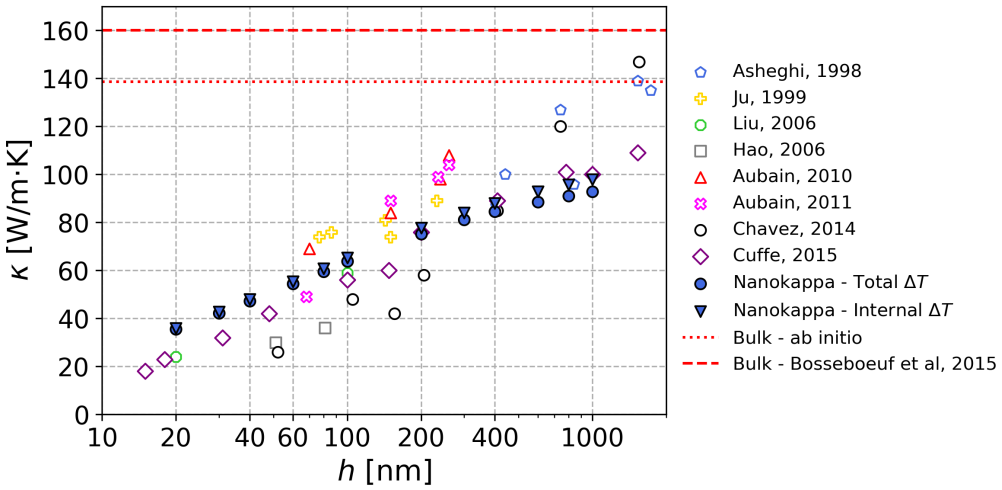

Figure 27: Simulation results for film in-plane conductivity compared with experimental data [15, 20, 13, 2, 3, 7, 9, 4].

3.5. Complex geometry

To show the capabilities of Nano-κ, the relatively complex mesh in Fig. 30 was simulated. The top and bottom facets were set as periodic, and temper-

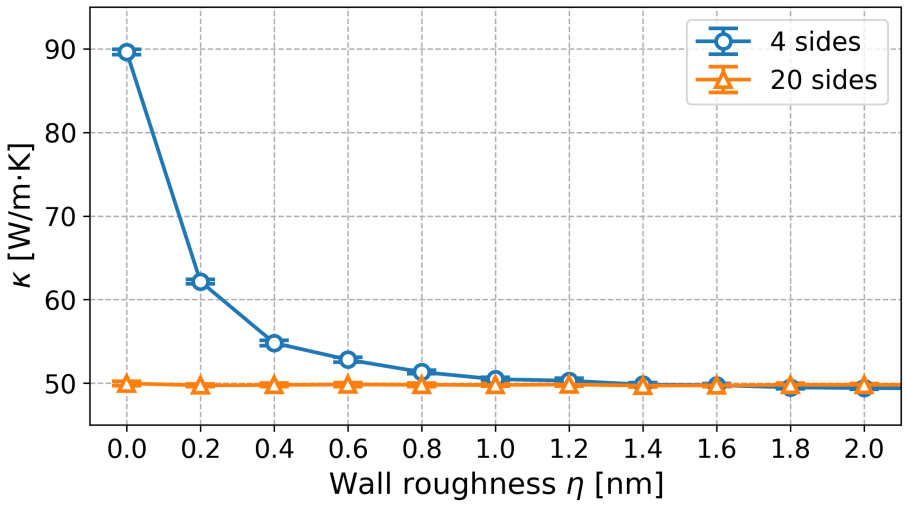

Figure 28: Comparison between thermal conductivity values calculated by Nano-κ for nanowires with square (4 sides) and polygonal (20 sides) cross-sections; length between thermostats at 302 K and 298 K is 2 µm.

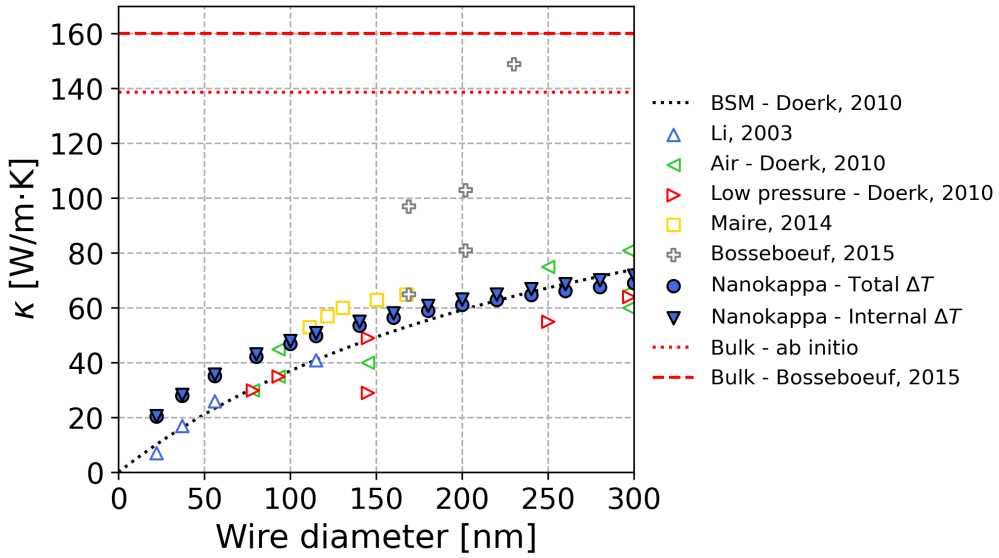

Figure 29: Simulation results for nanowire conductivity compared with experimental data referenced by Li et al. [18], Doerk et al. [12], Maire and Nomura [21], Bosseboeuf et al. [4].

atures were imposed on three facets, shown in blue. Facet 0 emits phonons at 302 K while facet 4 is at 298 K. The third reservoir at facet 8 had its temperature set in two different cases: a cold one (298 K) and hot one (302 K). The remaining facets were considered rough with $\eta = 0$ nm, making all reflections specular. The bounding box of the geometry measures 500 nm $\times$ 300 nm $\times$ 100 nm. For the subvolumes, a 18×9×1 grid layout was used, keeping only the subvolumes contained inside the geometry. For the grid layout, the conductivity is calculated locally for each connection between subvolumes, hence no global κ can be calculated.

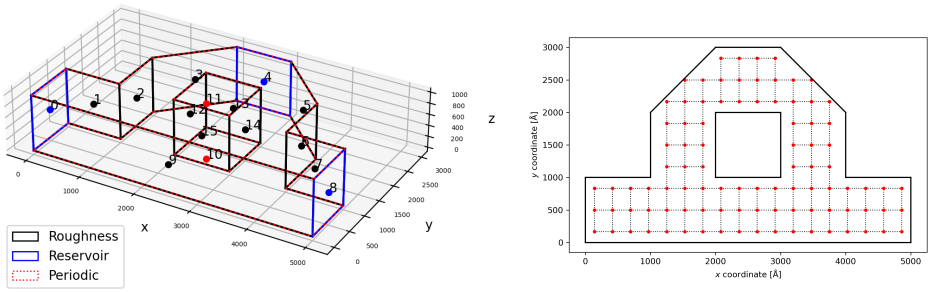

Figure 30: Complex geometry simulated. Left: Type of boundary conditions imposed on each facet. Right: subvolumes' reference points and connections.

Figure 3.5 shows the results after the system reaches steady state. Each plot is colored according to a different quantity. The arrows show the magnitude and direction of the heat flux. On the left, all plots refer to the cold case (two cold BCs, one hot BC), while the plots on the right refer to the hot case (one cold BC, two hot BCs).

The first plot is the temperature distribution. Because these simulations did not use interpolation of temperature, the borders of the subvolume can be seen clearly. The cold case has a fast decay of temperature on the left side, until it reaches around 300 K where the flux branches divide in two. Most of the heat flux goes straight to the right side, given that that is the path with less resistance. In the hot case, the temperature follows the symmetry of the domain and the boundary conditions, with the flux being redirected from the x direction to the y direction, leaving at the cold reservoir at the top.

A more refined view can be seen in the second plot, where it is shown the distribution of particles coloured according to their energy deviation from 300 K, $\delta e_{\mathbf{k}j} = \hbar\omega_{\mathbf{k}j}[n_{\mathbf{k}j} - n^0_{\mathbf{k}j}(300)]$. It is possible to see the modes responsible

for ballistic effects travelling through the straight portion of the geometry as isolated blue dots close to the hot side or red dots near the cold side.

To analyse the heat flux, the two last figures show particles colored according to their heat flux in x and y directions respectively, for each of the simulated cases. To better see this on the plot, the energy flux of each particle was calculated as a deviation to the local temperature (T_{local}): $\mathbf{\Phi} = \hbar\omega[n - n^0(T_{local})]\mathbf{v}$. Moreover, every particle with a calculated $|\mathbf{\Phi}_x|$ or $|\mathbf{\Phi}_y|$ of less than $2.5 \times 10^{-}3$ TeV/$\mathring{A}^2$·s was filtered out of the plots, so only the most significant particles, that carry energy are shown. This basically results in filtering the optical modes, the ones with shorter lifetimes that are often close to local equilibrium, and thus do not contribute much to $\mathbf{\Phi}$.

In the x direction, we can see the amount of particles that have a positive flux (in the $+x$ direction) for the cold case, concentrated in the bottom straight path. Outside of this path, there are also particles with significant $\mathbf{\Phi}_x$, but in equal amount for $-x$ and $+x$, and thus the arrows indicate that there is basically no flux horizontally. For the hot case, the flux is again symmetric, with the left side showing shades of red and yellow ($\mathbf{\Phi}_x > 0$) and the right side exhibiting shades of blue ($\mathbf{\Phi}_x < 0$). Below the square hole, particles in both senses meet and balance out with a local flux equal zero. Similarly, in the y direction, regions with balanced blue and yellow colored particles have near zero $\mathbf{\Phi}_y$. The figure also shows the phonons that travel upwards to the cold reservoir, in less extent for the cold case, but increasingly high as the particles approach the top end of the geometry. Such calculations and results visualization clearly point the ability of Nano-κ to handle complex geometries and extract relevant and specific outputs that help to understand thermal transport in complex nanodevices.

4. Future developments

Nano-κ was able to provide good estimates of thermal conductivity for several configurations and conditions, validating its results by comparison with experimental data. Nevertheless, we believe that more can be achieved in the future to improve their precision.

Such deviations between simulation and experimental can be due to several factors. First, any experimental data has uncertainties, which are often not reported. Second, the code does not account for the amorphous layer on the surface of the wire, for defects in the crystalline structure or for the eventual deformation of the nanowire when supported by the edges, which

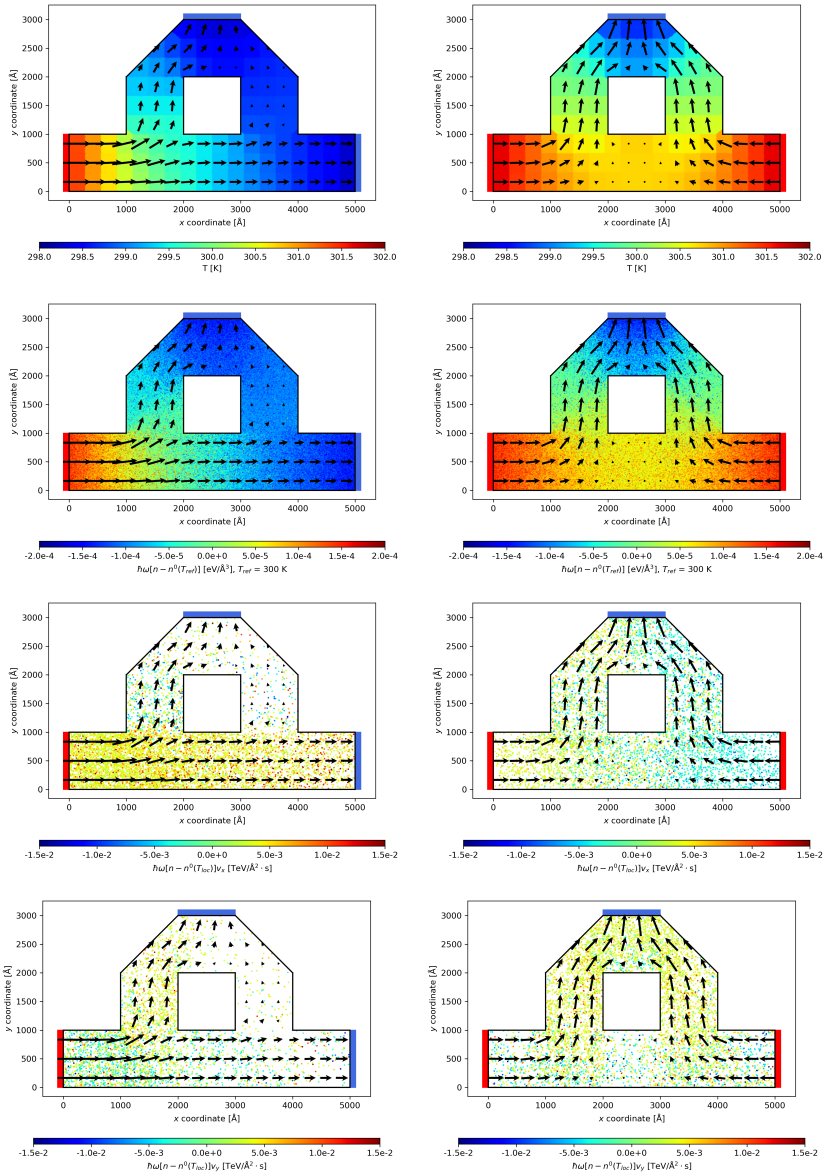

Figure 31: Color scatter plots for the two cases with complex geometry. In the left, the temperature of the reservoir at the right extremity is 298 K; on the right, it is 302 K. The arrows represent the heat flux magnitude and direction for each subvolume. The hot (red) and cold (blue) reservoirs' locations are shown as attached rectangles. From top to bottom: subvolume temperature, particle energy deviation ($\hbar\omega\delta n$) particle heat flux ($\hbar\omega\delta n\mathbf{v}$) in x and y directions.

can affect the conduction of heat. Such deviations can be also related to the input *ab-initio* data used in our model that may not fully correspond to synthesised materials. Beside, the boundary scattering model derived from Ziman [30] for acoustic phonons and generalised for the whole FBZ is likely incomplete. Presently, the reflections are considered to perfectly conserve energy, with all energy being kept in the simulated phonons (in average). In real nanostructures, this is probably not true, since there must be energy exchange with the amorphous superficial layer.

Other additions that may help to increase the generality of the code are:

- Transmission between materials;
- Heat flux as boundary condition;
- Energy exchange with the boundary;
- Automatic optimisation capabilities;
- Code parallelisation and acceleration.

Next iterations of the model should consider the above cited issues, and with this improve the accuracy of Nano-κ in comparison to experiments as well as its computational speed.

5. Conclusion

This paper brought the description of Nano-κ, a Python code dedicated to solve the Boltzmann Transport Equation for phonons in semiconducting nano-devices. The equations used by the algorithm and its structure were exposed. A sensitivity analysis was performed in order to analyse the effect of each simulation parameter on the final result. Thin films and wires were studied under different boundary condition settings and constitutive materials. All those calculations were compared with results in the literature. The results showed that satisfactory agreement can be achieved in all simple case situations were experimental data were available.

An example calculation with a more complex geometry was also discussed to show the flexibility and stability that Nano-κ can offer. Proposals for future works include the development of new boundary scattering models, modelling of phonon transport through interfaces between different materials and addition of optimisation routines. Under its presents form, the code

will be open through GitHub sharing in order to promote collaboration and improvements by interested users.

Acknowledgements

The authors thank the French National Research Agency (ANR) for the financial support by project LOR-AI (ANR-20-THIA-0010-01). High-performance computing resources were partially provided by the EXPLOR centre hosted by the University of Lorraine, additional HPC resources were partially provided by GENCI-TGCC and GENCIIDRIS (project # A0130913052).

Author contributions

B. H. Silva: conceptualization, formal analysis, investigation, methodology, software, visualization, and writing (original draft and review); D. Lacroix: conceptualization, formal analysis, investigation, methodology, funding acquisition, resources, supervision, validation, and writing (original draft and review); M. Isaiev: manuscript review and computational resources; L. Chaput: conceptualization, formal analysis, investigation, methodology, funding acquisition, resources, supervision, validation, and writing (original draft and review).

References

[1] M. Asheghi, K. Kurabayashi, R. Kasnavi, and K. E. Goodson. Thermal conduction in doped single-crystal silicon films. *Journal of Applied Physics*, 91(8):5079–5088, 03 2002. ISSN 0021-8979. doi: 10.1063/1.1458057. URL `https://doi.org/10.1063/1.1458057`.

[2] Max S. Aubain and Prabhakar R. Bandaru. Determination of diminished thermal conductivity in silicon thin films using scanning thermoreflectance thermometry. *Applied Physics Letters*, 97(25):253102, 12 2010. ISSN 0003-6951. doi: 10.1063/1.3527966. URL `https://doi.org/10.1063/1.3527966`.

[3] Max S. Aubain and Prabhakar R. Bandaru. In-plane thermal conductivity determination through thermoreflectance analysis and measurements. *Journal of Applied Physics*, 110(8):084313, 10 2011. ISSN 0021-8979. doi: 10.1063/1.3647318. URL `https://doi.org/10.1063/1.3647318`.

[4] Alain Bosseboeuf, Pierre Etienne Allain, Fabien Parrain, Xavier Le Roux, Nathalie Isac, Serge Jacob, Alexis Poizat, Philippe Coste, Seiffedine Maaroufi, and Arnaud Walther. Thermal and electromechanical characterization of top-down fabricated p-type silicon nanowires*. *Advances in Natural Sciences: Nanoscience and Nanotechnology*, 6(2): 025001, feb 2015. doi: 10.1088/2043-6262/6/2/025001. URL `https://dx.doi.org/10.1088/2043-6262/6/2/025001`.

[5] Carlo Bradac, Shuang Fang Lim, Huan-Cheng Chang, and Igor Aharonovich. Optical nanoscale thermometry: From fundamental mechanisms to emerging practical applications. *Advanced Optical Materials*, 8(15):2000183, 2020. doi: https://doi.org/10.1002/adom.202000183. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/adom.202000183`.

[6] Laurent Chaput, Jérôme Larroque, Philippe Dollfus, Jérôme Saint-Martin, and David Lacroix. Ab initio based calculations of the thermal conductivity at the micron scale. *Applied Physics Letters*, 112(3): 033104, 2018. doi: 10.1063/1.5010959. URL `https://doi.org/10.1063/1.5010959`.

[7] E. Chávez-Ángel, J. S. Reparaz, J. Gomis-Bresco, M. R. Wagner, J. Cuffe, B. Graczykowski, A. Shchepetov, H. Jiang, M. Prunnila, J. Ahopelto, F. Alzina, and C. M. Sotomayor Torres. Reduction of the thermal conductivity in free-standing silicon nano-membranes investigated by non-invasive Raman thermometry. *APL Materials*, 2(1): 012113, 01 2014. ISSN 2166-532X. doi: 10.1063/1.4861796. URL `https://doi.org/10.1063/1.4861796`.

[8] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2023. URL `http://www.blender.org`.

[9] John Cuffe, Jeffrey K. Eliason, A. A. Maznev, Kimberlee C. Collins, Jeremy A. Johnson, Andrey Shchepetov, Mika Prunnila, Jouni Ahopelto, Clivia M. Sotomayor Torres, Gang Chen, and Keith A. Nelson. Reconstructing phonon mean-free-path contributions to thermal conductivity using nanoscale membranes. *Phys. Rev. B*, 91:245423, Jun 2015. doi: 10.1103/PhysRevB.91.245423. URL `https://link.aps.org/doi/10.1103/PhysRevB.91.245423`.

[10] Brice Davier, Jérôme Larroque, Philippe Dollfus, Laurent Chaput, Sebastian Volz, David Lacroix, and Jérôme saint martin. Heat transfer in rough nanofilms and nanowires using Full Band Ab Initio Monte Carlo simulation. *Journal of Physics: Condensed Matter*, 30(49): 495902, October 2018. doi: 10.1088/1361-648X/aaea4f. URL `https://hal.archives-ouvertes.fr/hal-01906247`.

[11] Dawson-Haggerty et al. trimesh, 2023. URL `https://trimsh.org/`.

[12] Gregory S. Doerk, Carlo Carraro, and Roya Maboudian. Single nanowire thermal conductivity measurements by raman thermography. *ACS Nano*, 4(8):4908–4914, 2010. doi: 10.1021/nn1012429. URL `https://doi.org/10.1021/nn1012429`. PMID: 20731463.

[13] Z. Hao, L. Zhichao, T. Lilin, T. Zhimin, L. Litian, and L. Zhijian. *8th International Conference on Solid-State and Integrated Circuit Technology ICSICT'06 (IEEE, Piscataway, 2006)*, page 2196–2198, 2006.

[14] M. G. Holland. Analysis of lattice thermal conductivity. *Phys. Rev.*, 132:2461–2471, Dec 1963. doi: 10.1103/PhysRev.132.2461. URL `https://link.aps.org/doi/10.1103/PhysRev.132.2461`.

[15] Y. S. Ju and K. E. Goodson. Phonon scattering in silicon films with thickness of order 100 nm. *Applied Physics Letters*, 74(20):3005–3007, 05 1999. ISSN 0003-6951. doi: 10.1063/1.123994. URL `https://doi.org/10.1063/1.123994`.

[16] Juergen Riegel, Werner Mayer, Yorik van Havre. Freecad, 2022. URL `https://www.freecad.org/`.

[17] David Lacroix, Karl Joulain, and Denis Lemonnier. Monte carlo transient phonon transport in silicon and germanium at nanoscales. *Phys. Rev. B*, 72:064305, Aug 2005. doi: 10.1103/PhysRevB.72.064305. URL `https://link.aps.org/doi/10.1103/PhysRevB.72.064305`.

[18] Deyu Li, Yiying Wu, Philip Kim, Li Shi, Peidong Yang, and Arun Majumdar. Thermal conductivity of individual silicon nanowires. *Applied Physics Letters*, 83(14):2934–2936, 2003. doi: 10.1063/1.1616981. URL `https://doi.org/10.1063/1.1616981`.

[19] You Li, Guilei Wang, Mehdi Akbari-Saatlu, Marcin Procek, and Henry H. Radamson. Si and SiGe nanowire for micro-thermoelectric generator: A review of the current state of the art. *Frontiers in Materials*, 8, March 2021. doi: 10.3389/fmats.2021.611078. URL https://doi.org/10.3389/fmats.2021.611078.

[20] Wenjun Liu and Mehdi Asheghi. Thermal Conductivity Measurements of Ultra-Thin Single Crystal Silicon Layers. *Journal of Heat Transfer*, 128(1):75–83, 06 2005. ISSN 0022-1481. doi: 10.1115/1.2130403. URL https://doi.org/10.1115/1.2130403.

[21] Jeremie Maire and Masahiro Nomura. Reduced thermal conductivities of si one-dimensional periodic structure and nanowire. *Japanese Journal of Applied Physics*, 53(6S):06JE09, may 2014. doi: 10.7567/JJAP.53.06JE09. URL https://dx.doi.org/10.7567/JJAP.53.06JE09.

[22] A. A. Maradudin and A. E. Fein. Scattering of neutrons by an anharmonic crystal. *Phys. Rev.*, 128:2589–2608, Dec 1962. doi: 10.1103/PhysRev.128.2589. URL https://link.aps.org/doi/10.1103/PhysRev.128.2589.

[23] A.A. Maradudin, E.W. Montroll, and G.H. Weiss. *Theory of Lattice Dynamics in the Harmonic Approximation*. Solid state physics : advances in research and applications. Academic Press, 1963. ISBN 9780126077834.

[24] Sandip Mazumder and Arunava Majumdar. Monte Carlo Study of Phonon Transport in Solid Thin Films Including Dispersion and Polarization . *Journal of Heat Transfer*, 123(4):749–759, 01 2001. ISSN 0022-1481. doi: 10.1115/1.1377018. URL https://doi.org/10.1115/1.1377018.

[25] Ethan A. Scott, Christopher Perez, Christopher Saltonstall, David P. Adams, V. Carter Hodges, Mehdi Asheghi, Kenneth E. Goodson, Patrick E. Hopkins, Darin Leonhardt, and Elbara Ziade. Simultaneous thickness and thermal conductivity measurements of thinned silicon from 100 nm to 17 µm. *Applied Physics Letters*, 118(20):202108, 05 2021. ISSN 0003-6951. doi: 10.1063/5.0050888. URL https://doi.org/10.1063/5.0050888.

[26] Cheng Shao, Takuma Hori, and Junichiro Shiomi. P-trans: A monte carlo ray-tracing software to simulate phonon transport in arbitrary nanostructures. *Computer Physics Communications*, 276: 108361, 2022. ISSN 0010-4655. doi: https://doi.org/10.1016/j.cpc.2022.108361. URL `https://www.sciencedirect.com/science/article/pii/S0010465522000807`.

[27] Atsushi Togo, Laurent Chaput, and Isao Tanaka. Distributions of phonon lifetimes in brillouin zones. *Phys. Rev. B*, 91:094306, Mar 2015. doi: 10.1103/PhysRevB.91.094306.

[28] Atsushi Togo, Laurent Chaput, Terumasa Tadano, and Isao Tanaka. Implementation strategies in phonopy and phono3py. *J. Phys. Condens. Matter*, 35(35):353001, 2023. doi: 10.1088/1361-648X/acd831.

[29] Lina Wang, Mavd P.R. Teles, Ahmad Arabkoohsar, Haoshui Yu, Kamal A.R. Ismail, Omid Mahian, and Somchai Wongwises. A holistic and state-of-the-art review of nanotechnology in solar cells. *Sustainable Energy Technologies and Assessments*, 54:102864, 2022. ISSN 2213-1388. doi: https://doi.org/10.1016/j.seta.2022.102864. URL `https://www.sciencedirect.com/science/article/pii/S2213138822009122`.

[30] J.M. Ziman. *Electrons and Phonons: The Theory of Transport Phenomena in Solids*. International series of monographs on physics. OUP Oxford, 2001. ISBN 9780198507796.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: