Impact of the scattering physics on the power factor of complex thermoelectric materials

Patrizio Graziosi[1,2*], Chathurangi Kumarasinghe[1], Neophytos Neophytou[1]

Supporting information

A Method for the numerically efficient extraction of arbitrary 3D energy surfaces:

Here we describe the method we use to form the constant energy surfaces from an $E(\boldsymbol{k})$ and its validation. We adopt as a standard reference method the tetrahedron approach (R. M. Martin, Electronic Structure, Cambridge University Press, 2004) by means of a local Delaunay Triangulation (DT). The schematic in Fig. S1 shows the 2D version (for simplicity) of the 3D approach, the tetrahedra become here triangles, as in Ref. 20 of the main text.

The reciprocal unit cell is sampled by a regular mesh of 3D (2D in the figure) elements. For each energy (and each band) we scan all the mesh elements, numerically labelled by the first left corner point, shown by the green bullet at the left of Fig. S1, and check if the constant energy surface (contour in 2D) passes through it. To do this we check the energy of all the vertexes pairs (12 pairs in 3D) for the condition if at least one of these pairs consists of a vertex with energy above the value of interest (the energy surface value under consideration) and one below. If so, it means that the specific constant energy contour passes through the element under consideration. All the relevant elements are then triangulated. The $E(\boldsymbol{k})$ is taken to be linear between these points and thus a $\boldsymbol{k_i}$-point of energy E_i between the vertexes v1 and v2 is selected on each edge of the tetrahedron (triangle in the 2D schematic) as:

$$\boldsymbol{k}_i = \boldsymbol{k}_{\text{v1}} + (\boldsymbol{k}_{\text{v2}} - \boldsymbol{k}_{\text{v1}})\frac{E_i - E_{\text{v1}}}{E_{\text{v2}} - E_{\text{v1}}} \qquad \text{(s1).}$$

Each tetrahedron will have three or four points (according to if the vertexes above/below the certain energy are one or two) that define a surface computed with the Heron's formula. When the constant energy surface intersects two tetrahedra edges and we have four points the surface element is the average of the defined four triangles (the triangles in 2D have 2 points that define the length element). Such surface is the $\mathrm{d}A_{\boldsymbol{k}}$ described in the main manuscript and is used in the integration when computing the density of states. When we evaluate the anisotropic scattering rates (POP and IIS) we also need the coordinates of the points associated to a given surface (length in 2D) element, because the scattering rate depends on the wave vector difference between the final and initial states. We find the barycentre of the surface element (coordinates average) and this is the ***k***-point associated to the given surface (length) element, indicated by the green crosses in Fig. S1a.

We have also developed a simpler approach, which is at least ten times faster compared to the Delaunay Triangulating approach described above. In the simulation work of the main manuscript we used this scheme that we refer to as '***k***-scan'. In this method, we scan of all the ***k***-points and check only for their nearest neighbours along the mesh edges. If two points have energy one above and one below the value of interest, we select a ***k***-point along the edge of the mesh assuming a linear $E(\boldsymbol{k})$ between the two points, as above in Eq. s1. This point, indicated by the magenta cross in Fig. S1b in each element, is the ***k***-point at the energy of interest for the scattering rate evaluation. In this way we do not have any constant energy surface/contour element, but acquire just a collection of points on the energy surface of interest. We then assign an effective $\mathrm{d}A_{\boldsymbol{k}}$ surface element area value to each point to allow us to extract the density of states associated with that ***k***-point.

For this, after grouping the dispersion points on the constant energy surfaces, the surface in the neighbourhood of each ***k***-point is explored to detect its neighbours on the surface, in a radius of $\surd 2dk$ where dk is the distance between adjacent points in the initial mesh, the one used to calculate the bands. In the case where the ***k***-mesh discretization is different in the different coordinates, dk is the average of the different dk. Then, we calculate the average distance between the given point and its detected neighbours, $< \Delta k >$. The surface element associated to the ***k***-point is approximated by a circle of radius half the average distance to the neighbouring points, i.e. $\mathrm{d}A_k = \pi \left(\frac{<\Delta k>}{2}\right)^2$, see Fig. S1c. This method, that approximates the area of the surface element based on simple averaging of the nearest neighbour distances, rather that forming a full Tetrahedra mesh, is 10 to 15 times faster, but provides very comparable results. The validation of this scheme is presented in Fig. S2, where we show the density-of-states DOS for the cases of parabolic and non-parabolic bands, respectively. In Fig. S2 '***k***-scan' refers to this faster scheme, and 'DT' to the tetrahedron method. In both cases, for the isotropic parabolic isotropic (Fig. S2a) and the isotropic non-parabolic anisotropic (Kane's model, DOS $= \frac{\sqrt{2} m_{\mathrm{eff}}^{\frac{3}{2}}}{\pi^2 \hbar^3} \sqrt{E(1+\alpha E)}(1+2\alpha E)$, Fig. S2b), where in the anisotropic case $m_{\mathrm{eff}} = (m_{\mathrm{x}} m_{\mathrm{y}} m_{\mathrm{z}})^{1/3}$. Excellent agreement is achieved between the two methods, and compared to the analytic solution. In Fig. S2c we show the DOS(E) of the TiCoSb as calculated by

BoltzTraP, and as calculated by the '***k***-scan' and 'DT' methods, indicating also very good agreement.

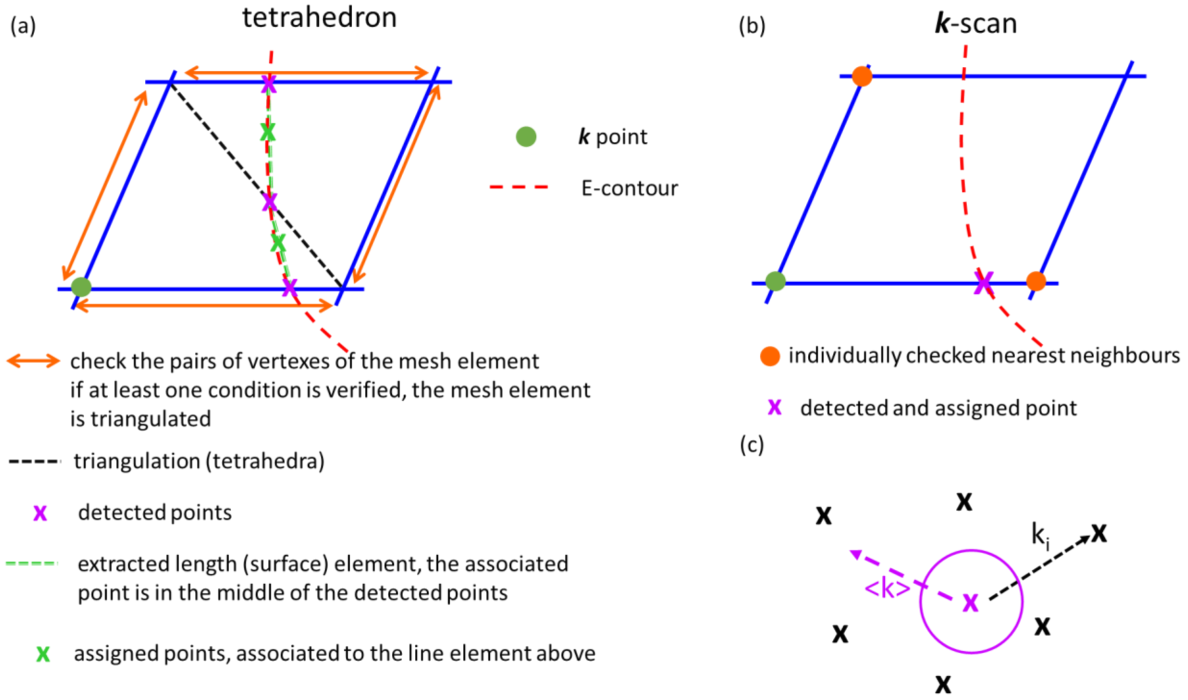
(a)
tetrahedron
k point
E-contour
check the pairs of vertexes of the mesh element if at least one condition is verified, the mesh element is triangulated
triangulation (tetrahedra)
x detected points
extracted length (surface) element, the associated point is in the middle of the detected points
x assigned points, associated to the line element above
(b)
k-scan
individually checked nearest neighbours
x detected and assigned point
(c)
x
k_i
<k>

Figure S1: 2D schematic of the procedure used to extract the ***k***(E), i.e. to compose the constant energy surfaces and assign a specific *k*-dependent DOS to each state. (a) The standard "tetrahedron" method (a) and the '***k***-scan' approach developed in this work (b). Here we depict 2D schematics for illustration clarity, but the work was performed in 3D. The blue lines represent an element of the initial regular 2D mesh used to compute the bands. The green point represents the "origin-reference" of the particular element. The red dashed line is the constant energy contour. In the tetrahedron method, if the mesh element is crossed by the contour, it is triangulated and the relevant points and length elements are extracted as described in the text. (b) In the '***k***-scan' approach only the nearest neighbours along the mesh are analysed and the detected relevant ***k***-point is the one assigned to the surface as described in the text. A corresponding effective surface element is then assigned as described in the text. (c) Here a conceptual scheme of the dA_k is presented where the specific point is in magenta, its neighbours in back, k_i is the distance between the point in magenta and the i-neighbour and <k> is the average distance. The surface element is represented by the magenta circle.

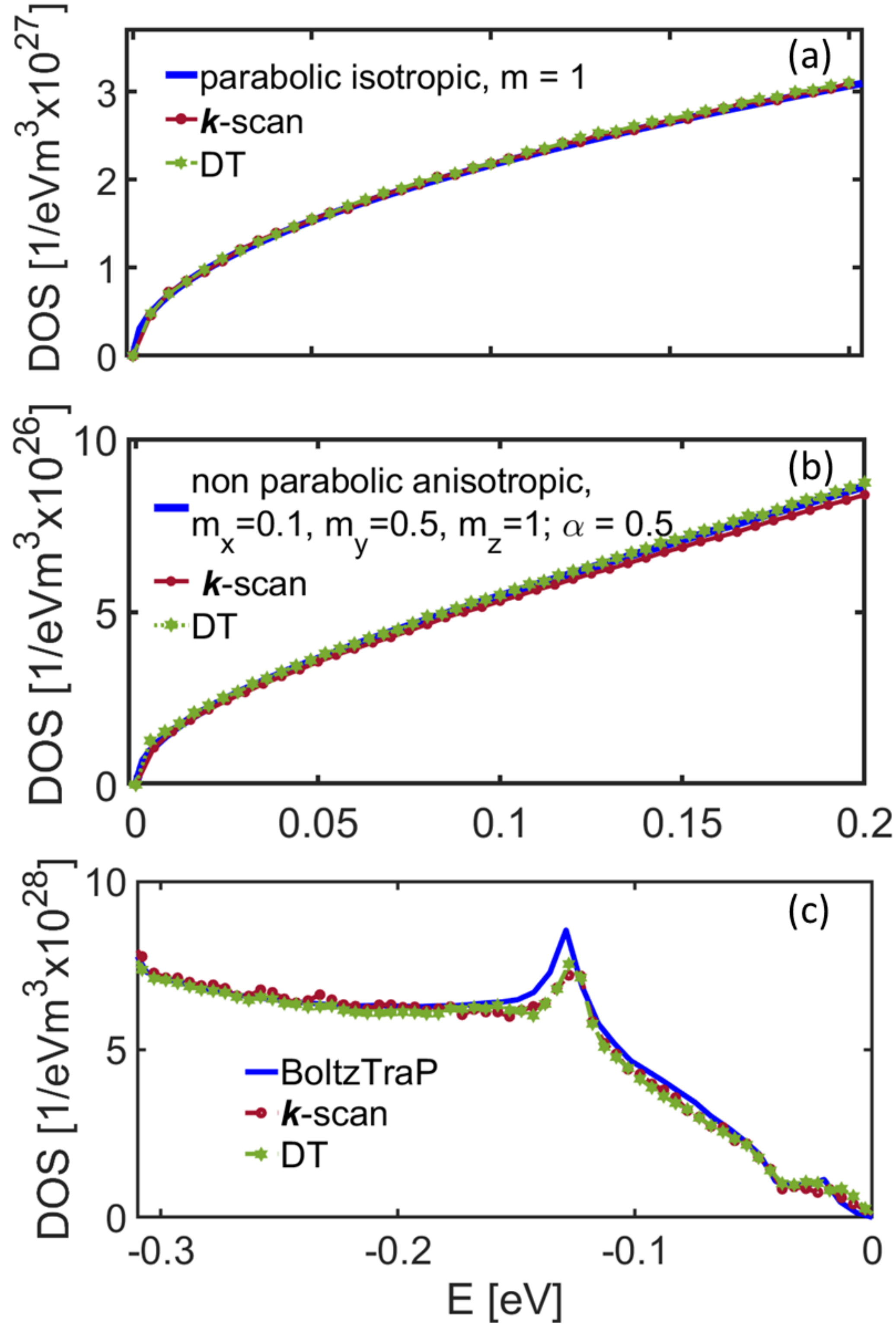

Figure S2: validation of the ***k***-scan approach versus the rigorous tetrahedron method (DT) and versus analytically known DOS – isotropic parabolic (a) and anisotropic non-parabolic (b) band – and versus BoltzTraP – the valence band DOS for TiCoSb is shown in (c). in (a) and (b) the masses m are in electron rest mass units and α is in eV^{-1}.

NOTE:

Mathematically, one can estimate an upper limit to the error introduced by this faster method, considering a plane with a regular discretization mesh. The area of an elemental square around a k-point is a_0^2, where a_0 is the mesh discretization. The average distance between the 8 nearest neighbours in a 2D surface is $a_0(1+\sqrt{2})/2$. The area of a circle half this radius is

$$\frac{\pi(3+2\sqrt{2})}{16} {a_0}^2 \sim 1.14{a_0}^2 \qquad (s2).$$

In other words, this approach in the case of uniformly spaced points on a planar 2D surface, brings a 14% error. On the other hand, the actual surface is not a plane and the points are not uniformly distributed on the surface so that the actual element on a surface will be somewhat larger. This reduces the error significantly.