I Parallelization of DFPT calculations

Calculations with the `PHonon` package are significantly more time intensive than `PWscf` calculations, so good parallelization is of the essence to make these calculations manageable.

I.1 Optimal parallelization parameters for DFPT calculations

The `PHonon` package offers the same three parallelization levels as the `PWscf` package, namely plane wave, k point and linear algebra parallelization. Furthermore parallelization across q points (so called image parallelization) can be employed, this will be discussed separately in sec. I.2.

I.1.1 k point parallelization

In a first step, the same k point parallelization benchmark as in sec. **??** is run. This is pictured in fig. I.1.

Interestingly, the results from sec. **??** is not reproduced here: the smallest pool size of 2 is not the fastest, but instead it is pool size 8.

I.1.2 Linear algebra parallelization

I.2 Image parallelization

Better introduction

When using image parallelization, Quantum ESPRESSO outputs a separate time report for every image, so one step is added to the analysis: The total runtime of a calculation is determined by the longest running image, so speedup will be calculated using that value, but another important measure to evaluate is variation of times between images. This is pictured in fig. I.4.

As the times between images don't vary much, good load balancing between images can be assumed for the silicon benchmarking system.

With the maximum time across images, speedup is then calculated, pictured in fig. I.4.

I.3 Conclusion: Parameters for optimal scaling

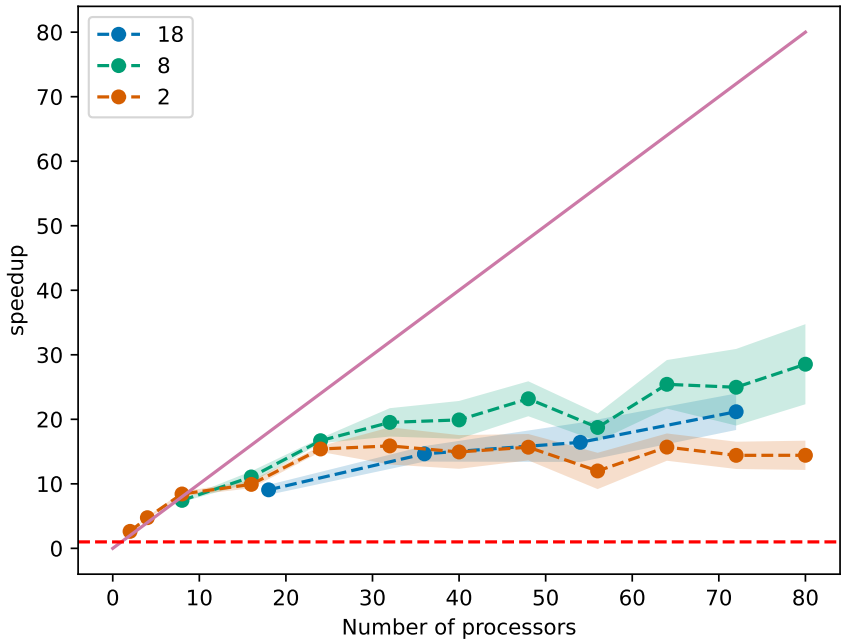

Figure I.1: *Scalability utilizing k-point parallelization for the Si benchmarking system with three sizes of processor pools,* QUANTUM ESPRESSO compiled with Intel oneAPI 2021.4, `nd 1`

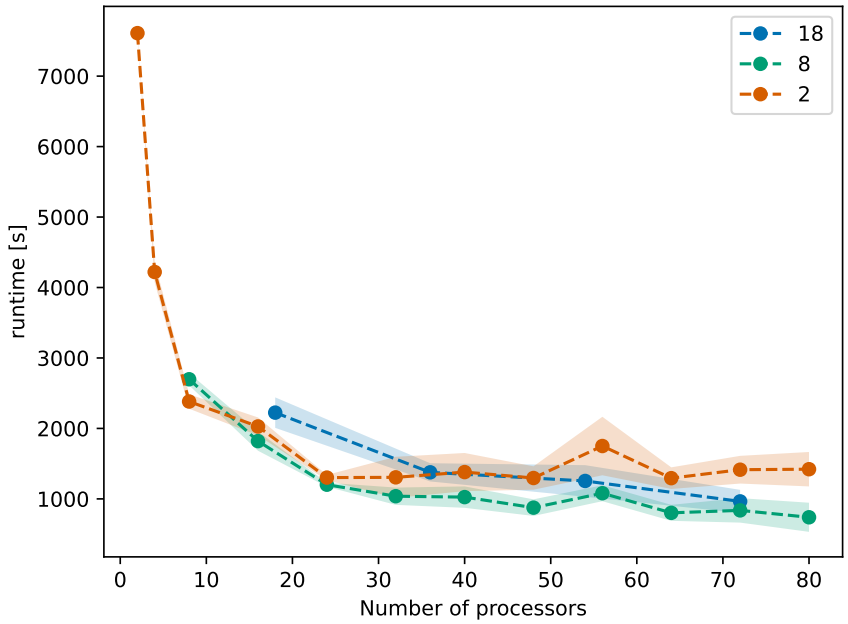

Figure I.2: *Absolute runtime for the scalability test utilizing k-point parallelization for the Si benchmarking system with three sizes of processor pools,* QUANTUM ESPRESSO compiled with Intel oneAPI 2021.4, `nd 1`

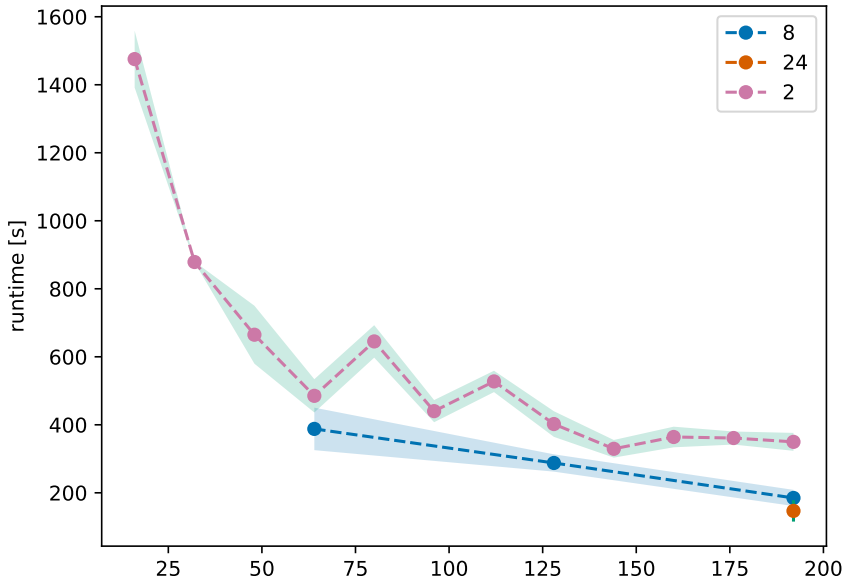

Figure I.3: *Average runtime across images for the scalability test utilizing image and k point parallelization on the Si benchmarking system with three values of* `ni`, QUANTUM ESPRESSO compiled with Intel oneAPI 2021.4, `nk, ni` chosen such that poolsize = 8, `nd 1`

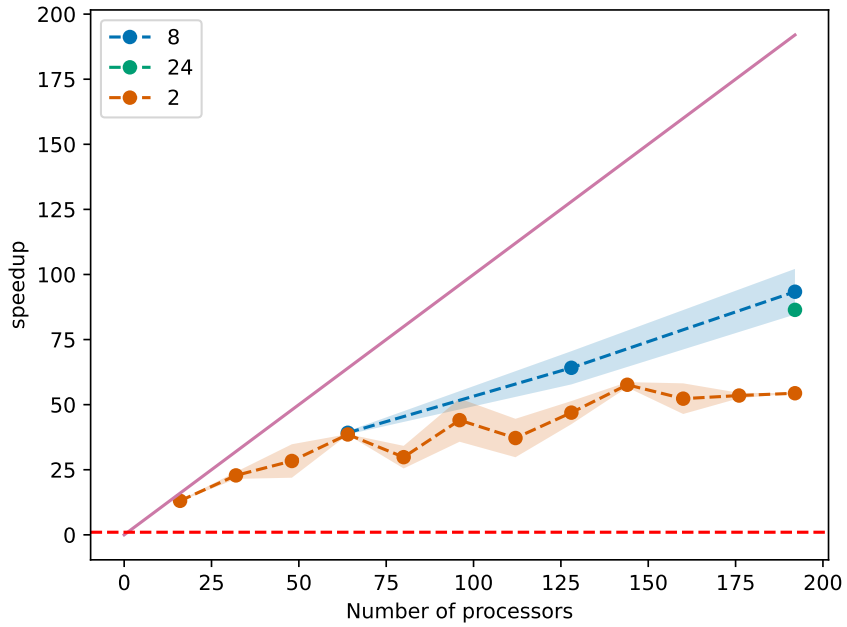

Figure I.4: *Speedup calculated from the longest running image for the scalability test utilizing image and k point parallelization on the Si benchmarking system with three values of* `ni`, QUANTUM ESPRESSO compiled with Intel oneAPI 2021.4, `nk, ni` chosen such that poolsize = 8, `nd 1`

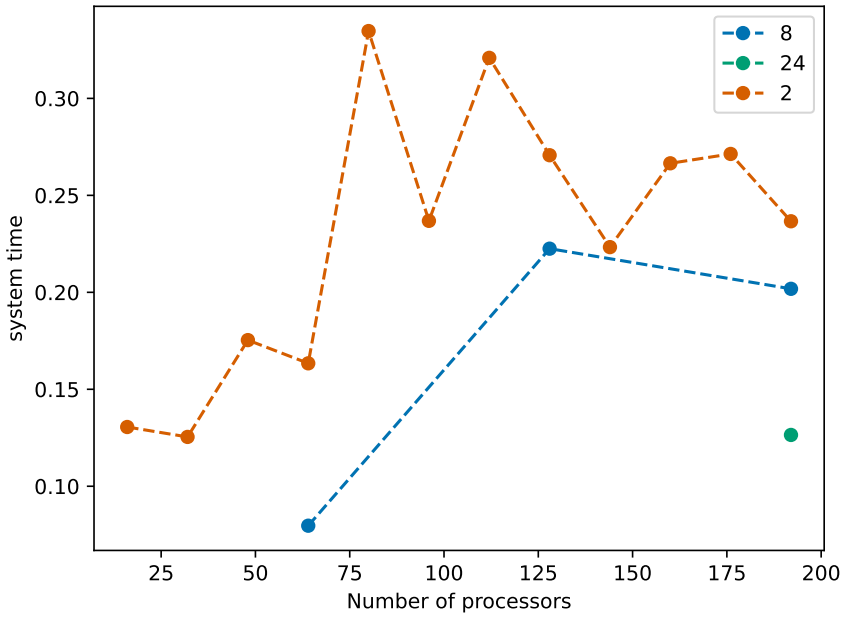

Figure I.5: *Wait time calculated from the longest running image for the scalability test utilizing image and k point parallelization on the Si benchmarking system with three values of* `ni`, QUANTUM ESPRESSO compiled with Intel oneAPI 2021.4, `nk, ni` chosen such that poolsize = 8, `nd 1`