Fighting Gravity Gradients with Gradient-Based Optimization at the Einstein Telescope

Patrick Schillings and Johannes Erdmann

RWTH Aachen University, III. Physikalisches Institut A, Aachen, Germany

Abstract

Gravity gradient noise in gravitational wave detectors originates from density fluctuations in the adjacency of the interferometer mirrors. At the Einstein Telescope, this noise source is expected to be dominant for low frequencies. Its impact is proposed to be reduced with the help of an array of seismometers that will be placed around the interferometer endpoints. We reformulate and implement the problem of finding the optimal seismometer positions in a differentiable way. We then explore the use of first-order gradient-based optimization for the design of the seismometer array for 1 Hz and 10 Hz and compare its performance and computational cost to two metaheuristic algorithms. For 1 Hz, we introduce a constraint term to prevent unphysical optimization results in the gradient-based method. In general, we find that it is an efficient strategy to initialize the gradient-based optimizer with a fast metaheuristic algorithm. For a small number of seismometers, this strategy results in approximately the same noise reduction as with the metaheuristics. For larger numbers of seismometers, gradient-based optimization outperforms the two metaheuristics by a factor of 2.25 for the faster of the two and a factor of 1.4 for the other one, which is significantly outperformed by gradient-based optimization in terms of computational efficiency.

1 Introduction

The observation of gravitational waves (GW) from binary black-hole mergers [1] with the LIGO [2] and Virgo [3] experiments confirmed the prediction of GWs by Einstein's theory of general relativity and opened up the field of GW astronomy. Since then, many more GW signals were measured with LIGO and Virgo, including GWs from binary neutron star mergers [4] and mergers of neutron stars and black holes [5]. The Einstein Telescope (ET) [6] is proposed as a next-generation GW detector. It is designed to significantly improve over the sensitivity of the LIGO, Virgo and KAGRA [7, 8] interferometers and aims to measure fainter signals and in particular signals at lower frequencies.

In order to suppress noise from seismic activity, the experiment is proposed to be placed underground and suspension systems will shield the interferometer mirrors from residual seismic displacement. Seismic waves traveling through Earth, however, also lead to tiny density fluctuations of the rock surrounding the mirrors. For low frequencies (below approximately 10 Hz), the resulting gravitational force is indeed a limiting factor for GW measurements [9, 10]. The interferometer cannot be shielded from this so-called "gravity gradient" or "Newtonian noise". However, the surrounding seismic wave field can be measured with seismometers in order to predict the movement of the interferometer mirrors—a technique that has already been successfully demonstrated at the Virgo experiment [11]. Since ET is to be built underground, boreholes need to be drilled for the seismometers and their positions hence need to be chosen with special care.

In Ref. [12], a Wiener filter was used to predict the Newtonian noise at the mirror positions based on the signals from the surrounding seismometers. An analytical model was constructed that describes the residual of the actual noise and the prediction for a single interferometer mirror. This model was then used as a cost function in the optimization of the seismometer positions using metaheuristics. This study was extended to the joint optimization of all mirrors in one interferometer endpoint using the maximum of the residuals as the cost function [13]. These studies showed that the optimization of the seismometer positions is possible up to a number of $N = 20$ seismometers and that the optimal positions depend on the assumed properties of the seismic field, in particular the frequencies that are used for optimization and the fraction of P-waves (primary/pressure waves) and S-waves (secondary/shear waves).

In order to reach optimal seismometer positions in the $3N$-dimensional parameter space, the optimizations in Refs. [12, 13] were repeated 100 times, finally picking the trial with the lowest residual. As a result, the

optimization is costly in terms of computing time and hence limited to $N \leq 20$. In addition, the results from the 100 trials displayed clear symmetries in the seismometer positions that, however, were not reflected in single trials, indicating that the metaheuristics may not have converged to the minimum of the cost function.

In this work, we study the prospects of gradient-based optimization techniques for the optimization of the seismometer positions for Newtonian noise suppression. We find that only one modification is necessary to reformulate the problem in a differentiable way and we implement the resulting cost function in JAX [14] to make use of JAX's support for automatic differentiation. We then use Adam [15] as an example of a well-established first-order optimization algorithm and compare its performance in terms of the achievable residual and an estimate of the optimization time. Such efforts to automate the optimization of experimental design with gradient-based methods have been studied in the areas of particle physics [16–22], muon tomography [23] and astrophysics [21, 24].

In Section 2, we introduce the cost function for Newtonian noise mitigation. We give details on the implementation of the gradient-based optimization approach in Section 3 and present our results in Section 4. In Section 5, we provide our conclusions.

2   Optimization of Seismometer Positions with Wiener Filters

With the Einstein Telescope to be built underground, the dominant type of seismic waves are body waves, of which two types exist: P- and S-waves. P-waves are longitudinal waves that lead to compression and dilation of rock inside Earth and thus directly change the density around the mirrors. They travel at a speed of sound of $c_P \approx 6 \, \mathrm{km \, s^{-1}}$. S-waves travel at a lower speed ($c_S \approx 4 \, \mathrm{km \, s^{-1}}$). They shift rock perpendicular to their direction of propagation and can lead to density fluctuations whenever there are boundaries of materials of different densities that are shifted closer to (or further from) the interferometer mirrors. P-waves, too, can have this influence, but in the direction of their propagation. As the effect of both waves differs, their relative contributions to the total wave field matter, which is given by the parameter p:

$$p = \frac{\langle \xi_\mathrm{P} \xi_\mathrm{P} \rangle}{\langle \xi_\mathrm{tot} \xi_\mathrm{tot} \rangle} \tag{1}$$

where $\xi_\mathrm{tot} = \xi_\mathrm{P} + \xi_\mathrm{S}$ is the total displacement field of the rock around the Einstein Telescope and ξ_P is the displacement field of only the P-waves. With this definition, P- and S-waves are assumed to be uncorrelated, i.e., we assume that there are no sources of seismic waves nearby.

To calculate the gravitational force on an interferometer mirror generated by the gravity gradients of a known displacement field of seismic waves, only linear operations are necessary (multiplication, integration). Thus, the Wiener filter [25] can be used to predict the mirror acceleration from the displacement measured by a finite number N of seismometers in all three spatial directions, as it is the optimal linear filter for a given frequency f. Under the assumptions of isotropic seismic waves, a homogeneous medium and plane waves, the Wiener filter in frequency space can be calculated analytically [26]. The resulting residual of the actual Newtonian noise on a mirror i and its Wiener filter prediction is given by:

$$R_i(\omega) = \frac{\langle |s(\omega) - \tilde{s}(\omega)|^2 \rangle}{\langle |s(\omega)|^2 \rangle} = 1 - \frac{\vec{c}_\mathrm{sd}^\dagger \cdot C_\mathrm{dd}^{-1} \cdot \vec{c}_\mathrm{sd}}{c_\mathrm{ss}} \tag{2}$$

where $s(\omega)$ is the Fourier transform of the signal (the Newtonian noise at the mirror i) at frequency $\omega = 2\pi f$, $\tilde{s}(\omega)$ is the prediction of the signal from the Wiener filter, C_dd is the cross power spectral density of the data (seismometer displacement measured in 3D), $\vec{c}_\mathrm{sd}$ is the cross power spectral density of the data vector with the signal, and c_ss is the power spectral density of the signal. The seismometer correlation C_{dd} contains an additional term on its diagonal for the self-noise of the seismometers: the signal-to-noise ratio SNR that is given by $\mathrm{SNR} = \sqrt{c_\mathrm{ss}/c_\mathrm{nn}}$, where c_nn is the power spectral density of the seismometer noise. We assume the noise of different seismometers and different channels of the same seismometer to be uncorrelated. The square root of the residual is an estimate of the remaining effect from Newtonian noise in the GW amplitude. Varying the positions of the seismometers changes the cross-power spectral densities in Eq. (2), hence the information that the Wiener filter can use, and consequently the quality of the noise mitigation. When optimizing the seismometer positions, the goal is to minimize the residual with respect to the $3N$ seismometer coordinates. So in total, the parameters of the problem consist of the number of seismometers N, the frequency f chosen to calculate the residual, the signal-to-noise ratio SNR and the abundance of P-waves p.

The Einstein Telescope is designed to consist of separately optimized interferometers for low- and high-frequency GWs, where only the test masses for the low-frequency part are expected to be significantly affected by Newtonian noise. For close-by test masses, their Newtonian noise is expected to be correlated. In Ref. [13], the

maximum of the residuals of the individual mirrors from Eq. (2) was chosen as the objective function for optimization:

$$R = \max_i(R_i) \qquad (3)$$

While different design options for the Einstein Telescope have been proposed [6, 27], here we study the triangle configuration, where each corner comprises four mirrors from the low-frequency interferometer. We use the same mirror geometry as in Ref. [13], where the four mirrors are separated by an angle of $60°$ and at a distance from the corner of $64.12\,\mathrm{m}$ and $536.35\,\mathrm{m}$, respectively. While numerical values in the optimization would change when repeating the studies for an L-shaped geometry, we expect the conclusions about the optimization methods to remain valid.

3   Optimization Algorithms and Formulation of a Differentiable Loss

Previous studies [12, 13, 28] made use of metaheuristics to optimize the seismometer positions, including basin hopping [29], particle swarm optimization (PSO) [30] and differential evolution (DE) [31]. In this work, we compare the gradient-based optimization with Adam to the results from DE, as this optimizer was selected for the most recent optimizations [13]. In addition, we compare to optimizations with PSO, as we find good performance at lower computational costs compared to DE. In the following, we provide a brief description of the different optimization algorithms and then discuss the differentiable formulation and implementation of the residual cost function.

For PSO, one chooses a number of swarm particles n_{swarm}. Each particle i is assigned a position $\vec{x}_i$ in the space of optimization parameters within given boundaries. In our case, this is the $3N$-dimensional space of the coordinates of the N seismometers. The objective function is evaluated at each particle position. Each particle remembers the lowest function value $\vec{p}_{\mathrm{best}}$ it has encountered (personal best). In each step t, each particle is assigned a velocity $\vec{v}_i$ that is calculated in the following way:

$$\vec{v}_i(t + 1) = \omega\vec{v}_i(t) + r_1 C_c(\vec{p}_{\mathrm{best},i}(t) - \vec{x}_i(t)) + r_2 C_s(\vec{g}_{k,\mathrm{best},i}(t) - \vec{x}_i(t)) \qquad (4)$$

where r_1 and r_2 are random numbers between 0 and 1 that are re-sampled in every step, and $\vec{g}_{k,\mathrm{best},i}$ is the lowest function value that all of the k nearest neighbors to particle i have encountered up to step t. The tuneable parameters of the PSO algorithm are ω ("inertia"), C_c ("cognitive parameter"), C_s ("social parameter") and k. After each step, the particle positions are updated: $\vec{x}_i(t + 1) = \vec{x}_i(t) + \vec{v}_i(t + 1)$.

The algorithm stops and returns $\vec{g}_{n_{\mathrm{swarm}},\mathrm{best}}(t_{\max})$ after a given number of steps $t_{\max}$ or, with early stopping, after $\vec{g}_{n_{\mathrm{swarm}},\mathrm{best}}$ has not changed more than a small tolerance value in the last P steps, with the patience parameter P. We use the PSO implementation in pyswarms[1] [32] with $n_{\mathrm{swarm}} = 800$, $C_c = 1.5$, $C_s = 2$ and $\omega = 0.1$ in a ring topology with $k = 80$. These values, except for n_{swarm}, were chosen in a random search that showed overall little dependency on the parameter values for the example of $f = 1$ Hz, $N = 3$, SNR $= 15$ and $p = 0.2$. A tolerance of $1 \cdot 10^{-3}$ is used with a patience of 20 steps and a maximum of 1000 steps.

For DE, one also chooses a number of positions $\vec{x}_i$ in the $3N$-dimensional space, called the population size n_{pop}. Again, the personal best $\vec{p}_{\mathrm{best},i}$ and global best $\vec{g}_{\mathrm{best},i}$ values of the objective function are saved in each step. To calculate new positions, for each position from the population, three distinct other population members $\vec{x}_a$, $\vec{x}_b$ and $\vec{x}_c$ are drawn. The new positions are then calculated as follows:

$$x'_{i,j} = \begin{cases} x_{i,j} & \text{if } r_j < p_{\mathrm{Cross}} \\ x_{a,j} + F \cdot (x_{b,j} - x_{c,j}) & \text{else} \end{cases} \qquad (5)$$

where j enumerates the dimensions of the position vectors and r_j is a distinct random number between 0 and 1. The tuneable parameters of the algorithm are the crossover probability p_{Cross}, which should be chosen between 0 and 1, and the differential weight F, which is typically set between 0 and 2, but can also be randomly generated for each generation of new positions. The new positions $\vec{x}'_i$ are accepted and replace $\vec{x}_i$ if their function value is equal to or smaller than that of the original position; otherwise, $\vec{x}_i$ remains a member of the population. Again, the final result is the global best after $t_{\max}$ or early stopping. We use the DE implementation in SciPy[2] [33] with a population size of 65 per dimension (so $n_{\mathrm{pop}} = 3N \cdot 65$), $p_C = 0.75$ and F drawn between 0 and 1.5 in each generation. We use random initialization, a tolerance of $1 \cdot 10^{-3}$ and a maximum number of steps of 4500.

Both algorithms are metaheuristics that search the parameter space within given boundaries, keep testing new positions that are methodically chosen based on the previous positions and, at some point, settle on the lowest

---

[1]pyswarms.single.global_best.GeneralOptimizerPSO()
[2]scipy.optimize.differential_evolution()

function value they have seen in the process. They can efficiently cover large parameter volumes in high-dimensional parameter spaces and do not require any assumptions on the objective function. In particular, they can also be used for non-differentiable and non-continuous functions. However, the higher the dimensionality of the parameter space, the lower the coverage of test evaluations for a fixed population size. In general, PSO and DE are not guaranteed to converge to local or global minima, in particular for high-dimensional optimization problems.

Adam, on the other hand, starts from a single initial position $\vec{x}_0$ that is gradually improved. As a variant of stochastic gradient descent, it is frequently used in the training of neural networks, where the mean of the objective function over a small number of training samples is used to calculate the gradients. In our use case, however, we use Adam to calculate gradients of a fixed cost function without a stochastic component. It is based on the gradient $\vec{g}(\vec{x})$ with $g_i(\vec{x}) = \frac{\mathrm{d}f}{\mathrm{d}x_i}(\vec{x})$. In every step, it updates the position according to

$$\vec{x}_t = \vec{x}_{t-1} - \alpha \frac{\vec{m}_t}{\sqrt{v_t} + \varepsilon} \tag{6}$$

where the parameter α is the "learning rate" and ε is a very small parameter that prevents division by zero. The value $\vec{m}_t$ is a weighted average of the gradients from past steps, resulting in an inertia in the evolution of $\vec{x}$. Similarly, v_t is the weighted squared derivative:

$$\vec{m}_t = \frac{\beta_1}{1 - \beta_1^t} \vec{m}_{t-1} + \frac{1 - \beta_1}{1 - \beta_1^t} \vec{g}(\vec{x}_t) \tag{7}$$

$$v_t = \frac{\beta_2}{1 - \beta_2^t} v_{t-1} + \frac{1 - \beta_2}{1 - \beta_2^t} \vec{g}^2(\vec{x}_t) \tag{8}$$

The parameters β_1 and β_2 are tuneable exponential decay rates for the two moment estimates. In each step, $f(\vec{x})$ is saved. As before, the algorithm is stopped after a certain number of steps (or with early stopping) and then returns the lowest function value that it encountered.

Adam requires $f(\vec{x})$ to be differentiable, as it makes use of its gradients. Due to its widespread application in deep learning, it is well understood and has proven applicable to a large variety of tasks. With the automatic differentiation capabilities of JAX, calculating the gradient becomes very efficient, even in high dimensions. The downside of Adam as a deterministic optimizer is that it may converge to local minima in the vicinity of its initial position.

Carefully choosing the initial position for Adam may hence be important, so we do not only try random initialization, but we also initialize Adam with seismometer positions PSO or DE. This is motivated by the fact that the metaheuristics are efficient in testing the whole parameter space, while Adam uses the objective function's gradient to converge to a nearby local (or global) minimum.

To make use of automatic differentiation in JAX, we have rewritten the code from Ref. [13] in JAX, making use of JAX's just-in-time compilation for computational efficiency. While this was mostly straightforward to implement, it required one important change compared to previous optimizations: to make the cost function differentiable, we replaced the max()-operation that combines the individual residuals of the different interferometer mirrors in Eq. (3) with a differentiable operation, for which we chose the mean of the four mirror residuals.

We use the implementation of Adam in `optax`[3] [34] with a learning rate[4] of $1\,\mathrm{m}$ and the default values of $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\varepsilon = 1 \cdot 10^{-8}\,\mathrm{m}^{-1}$. Early stopping with a tolerance of $1 \cdot 10^{-3}$ and a patience of 500 steps is used, while the maximum number of steps is 10000.

In a first step, the optimization is performed with DE, PSO and Adam using random initialization. Separate optimizations are done for the number of seismometers, N, varying between 1 and 70 in steps of 1. In Refs. [12, 13], the optimization with DE was repeated 100 times, picking the best optimization run. To achieve comparable results to these previous optimizations, we repeat the optimization with PSO 100 times with different initial values, also picking the best optimization run. We choose PSO instead of DE, as it is significantly faster than DE, and we limit the repeated runs of PSO for $N > 20$ to $N = 25$ as well as $N = 30$ to $N = 70$ in steps of 10. In a second step, optimization with Adam is performed using the results of the single PSO and DE runs as initialization, as well as—for a few selected values of N—using the best optimization from the 100 runs of PSO as initialization. We study the optimal positions for frequencies of 10 Hz and 1 Hz, which are example frequencies at the very upper and lower boundaries at which Newtonian noise is expected to be relevant for the Einstein Telescope. We assume an SNR of 15 and a fraction of P-waves, p, of 0.2, which is a benchmark that was thoroughly studied previously [13].

---

[3] `optax.adam()`
[4] The parameters $\vec{x}_i$ are the x-, y- and z-coordinates of the seismometer positions in meters.

4 Results

The results of the optimization at a frequency of $f = 10$ Hz are shown in Figure 1. The parameter boundaries for PSO and DE are chosen as $\pm 1000\,\text{m}$ in x- and y-direction and $\pm 300\,\text{m}$ in z-direction, with the origin of the coordinate system in the interferometer corner. Shown is the square root of the residual as a function of the number of seismometers N. In previous optimizations [12, 13], the number of seismometers was limited to $N \leq 20$, motivated by the cost of drilling one borehole per seismometer. We study also larger values of N in order to quantify the achievable gains in noise mitigation, as future studies may investigate the potential for optimizing the positions of seismometers that can be more cheaply deployed, such as in the interferometer tunnels.

All optimizations in Figure 1 show the same general trend: they start at a residual close to 1, that is reduced when seismometers are added. The more seismometers get deployed, the more information can be used by the Wiener filter to predict the Newtonian noise at the interferometer mirrors, to separate the effects of P- and S-waves, and to reduce the impact of the self-noise of the seismometers. The information gain per seismometer is highest when only a small number of seismometers is already deployed. Random initialization and the intrinsic randomness of the metaheuristics result in fluctuations around this general trend. These fluctuations can be reduced, for example, by repeated runs of the algorithm, comparing the single PSO run with the best of 100 PSO runs. Adam with random initialization shows particularly large fluctuations, as its outcome is solely determined by the initial seismometer coordinates.

Comparing single runs of the three optimization algorithms with random initialization, the two heuristic algorithms yield similar results for a small N but deviate around $N = 20$ with DE reaching ultimately better residuals for the chosen values of the hyperparameters. We note that we did not perform a separate hyperparameter optimization for every value of N. In particular, the number of swarm particles in PSO is constant for all values of N, while the population size in DE increases with N. The results from Adam with random initialization fluctuate strongly and are generally worse than the PSO and DE results up to $N \approx 50$, with only a few exceptions. For larger values of N, Adam with random initialization often outperforms PSO and DE, which indicates that the exact initialization for large N is not as crucial as for lower values.

Using the results of PSO and DE as initial positions for Adam has no effect for very small values of N. This means that in low dimensions, the metaheuristics are able to find values that are close to the bottom of a local minimum (or maybe even global minimum). Between $N \approx 5$ and $N \approx 15$, the metaheuristics find good values. Running Adam initialized with these results leads to a slight improvement, but it is often still trapped in the local minimum that was found with the corresponding metaheuristic. For larger values of N, the initialization of Adam with the metaheuristics leads to significant improvements in the optimization. The optimization leads to seismometer positions that are sometimes quite far away from the positions that the metaheuristics had found, resulting in average improvements of factors of 2.25 for PSO and of 1.4 for DE (cf. the upper ratio plot in Figure 1). The results after optimization with Adam only differ by a few percent between initialization with PSO or DE, with PSO occasionally yielding the better initialization for Adam. This means that it is not very important which metaheuristic is used to provide the initial values for Adam.

In Figure 1, we also show the results of 100 independent optimization runs of PSO, i.e., with the same hyperparameters but different initial values, choosing the best of all runs as the optimized values. This is comparable to the method that was used in Ref. [13] but with PSO instead of DE. As expected, this leads to fewer fluctuations in the optimized results. Again, for small N, no significant improvement is visible because already single runs of PSO or DE yield good solutions. Between $N \approx 5$ and $N \approx 15$, this method outperforms all of the single runs of the different optimizers. For $N = 8$ and $N = 9$, we tested if there is still potential for improvement by using Adam initialized with this best solution of these 100 PSO runs. Indeed, we find a few percent of improvement (cf. the lower ratio plot in Figure 1). For values of N between 15 and 20, running PSO 100 times yields comparable results to PSO+Adam, but for larger values of N, it is outperformed by the single runs of Adam initialized by PSO or DE by a factor of almost 2. Further optimizing the best of the 100 PSO runs with Adam does not lead to better results than running Adam initialized with a single PSO run, as tested for $N = 40$.

In Figure 2, we show the time that was required to calculate the points shown in Figure 1 on our institute's computing cluster. The times that we recorded for each run may vary due to the performance of each cluster node and by how many jobs run in parallel on each node, as is visible by the fluctuations of the run times for each optimization strategy. However, we verified that the general trends and statements in Figure 2 hold by performing a second independent full optimization run.

As expected, the computing time generally increases with N, i.e., with the number of parameters of the optimization task, but the time for the different methods grows differently fast. With our choice of hyperparameters, PSO runs quickest for high N. DE takes about 20 times longer than PSO but also yields better results. When Adam is initialized with either PSO or DE, it runs much faster than with random initialization, as the criteria
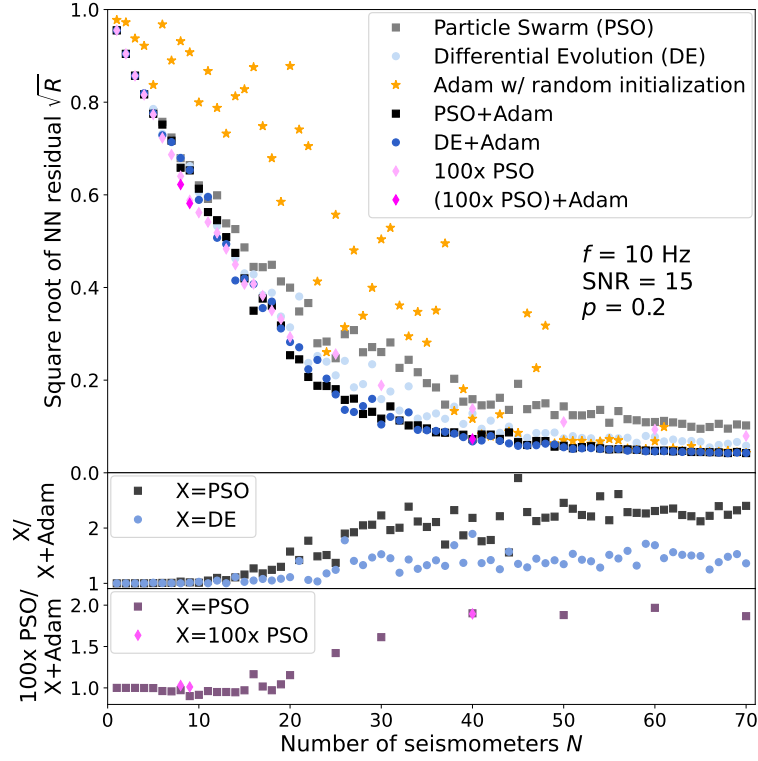
Figure 1: $\sqrt{R}$ for Newtonian noise mitigation as a function of N for $f = 10$ Hz, $p = 0.2$ and SNR = 15. The optimization was done with PSO, DE, Adam with random initialization and Adam initialized with PSO and DE, respectively. Also shown is the minimum value for 100 distinct runs of PSO (for $N > 20$, only $N = 25, 30, 40, 50, 60, 70$ were studied) with a few of them also used as initial values for optimization with Adam. The bottom panels show the ratio of the improvement in $\sqrt{R}$ obtained when initializing Adam with single runs of PSO or DE and with a single PSO run or 100 distinct runs of PSO, respectively.
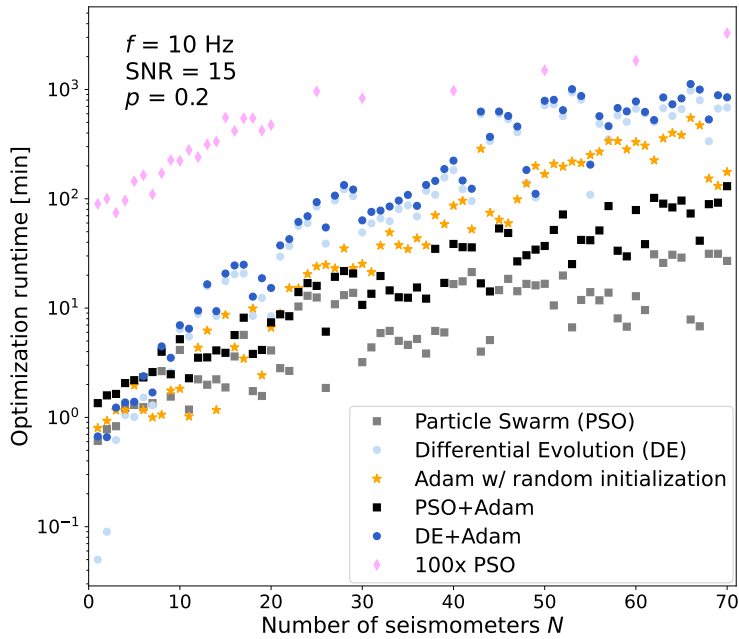
Figure 2: Time consumed for the optimization on the computing cluster by PSO, DE and Adam with random initialization as well as Adam initialized by single runs of PSO and DE and the sum of 100 PSO runs.

for early stopping are met faster given the good choice of initial parameter values. Even when combining the times of PSO and Adam, it is still below the time for an optimization with Adam with random initialization as well as lower than DE, while giving better residuals. Running PSO 100 times may yield slightly better results at moderate N, but can take a factor of 100 longer than the combined optimization of PSO and Adam for these numbers of seismometers.

The evolution of the residual during the optimization with Adam is shown in Figure 3 for $N = 6$ and $N = 40$ for the initialization with several distinct PSO runs. The optimization procedure stops after the curve is flat for a while due to early stopping. For $N = 6$, one can see that Adam improves the PSO solutions by about a percent. It is striking that the final solution often depends on the initial positions from PSO and follows the gradient to the bottom of a local minimum. This can be the same local minimum (or at least one with the same value) for different PSO pre-optimizations. For $N = 40$, the optimization with Adam provides significant improvements for Newtonian noise mitigation and quickly brings the different PSO solutions down to half of the initial residuals. The spread of the final solutions is small compared to the obtained improvements, and for such large values of N, it is hence not guaranteed that a better PSO initial residual will result in a better optimization with Adam.

Figure 3 suggests that Adam descends into distinct local minima from the initial PSO results. We further investigate this for $N = 6$ and ask whether these local minima correspond to specific geometries of the seismometer positions. Residuals of 100 PSO runs before and after further optimization with Adam are shown in Figure 4. Considering only the results after optimizing with PSO (light squares), they mainly are in the range between $\sqrt{R} \approx 0.73$ and $\sqrt{R} \approx 0.76$. After running Adam initialized with these PSO results (dark squares), the resulting residuals are not only lower on average but also many optimization runs result in very similar residuals. We interpret these values as local minima that Adam descends into from the original PSO positions. We also note that the amount of improvement that can be obtained from the optimization with Adam in general cannot be predicted from the PSO result alone.

Figure 5 shows the optimized sensor positions of repeated runs of PSO and PSO+Adam in the xz- and xy-planes for $N = 6$. In Figure 5 (a) and (c), the 6 positions from the 100 distinct runs of PSO (light squares from Figure 4) are shown. As discussed in Ref. [13], this shows the symmetries of the task[5]. While a symmetry is visible in the ensemble of the optimized positions, which corresponds to the symmetry of the geometry of the task, the run with the lowest residual among all 100 runs (orange points) does not show this overall symmetry itself.

In Figure 5 (b) and (d), the optimized positions are shown after PSO+Adam, this time only for runs that result in $\sqrt{R} \approx 0.7384$, which represent the runs from the prominent line of dark squares in Figure 4. We identified four distinct solutions among these runs and used different colors and symbols to mark the six seismometers that belong to each solution. One can see that all solutions can be obtained from a single one by mirroring along the symmetry planes of the problem, namely the $z = 0$ and $y = 0$ planes in this coordinate system. We interpret this as Adam being able to descend to the local minima of the cost function better than PSO, so that its optimized results very clearly show the symmetries of the optimization task, while the PSO results are smeared around the best positions because they do not descend all the way down to the bottom of the minimum.

While $f = 10$ Hz is at the upper end of the relevant frequency range for Newtonian noise, we also ask how Newtonian noise mitigation with the Wiener filter performs for low frequencies and pick $f = 1$ Hz as a very low example frequency. When we optimize for $f = 1$ Hz, we find that Adam converges to seismometer positions that are much further away from the mirrors than for $f = 10$ Hz. This is expected because the wavelength for 1 Hz and $c_P \approx 6\,\mathrm{km\,s^{-1}}$ ($c_S \approx 4\,\mathrm{km\,s^{-1}}$) is as large as 6 km (4 km) for P-waves (S-waves). Since the analytical model for Newtonian noise mitigation based on the Wiener filter assumes that the rock is isotropically distributed around the mirrors, this model fails to take into account that the Einstein Telescope is expected to be built approximately 300 m underground, so that the optimized seismometer positions must be restricted to a maximum distance in z-direction from the mirrors. While PSO and DE require that a search volume is defined for the optimization, this is not the case for Adam.

To prevent this, we study two methods. One method is to simply clip Adam's optimization results to the corresponding constraint by letting it finish its optimization and in the resulting positions, replacing every z-coordinate that has $|z| > 300$ m with $z = \pm 300\,\mathrm{m}$.[6] The other method is to multiply a differentiable function

---

[5]We note that the optimization in Ref. [13] was performed with the maximum of the residuals among the four mirrors in one corner and not for their mean.

[6]It is known that the rock is not homogeneous along the full depth of 300 m, i.e. the cross correlations of seismometer signals with the Newtonian noise at the mirrors are expected to vary as a function of seismometer depth. However, for simplicity, we assume that the rock is homogeneous up to $z = 300$ m and study how to effectively confine optimization results with Adam to a certain phase space. The same procedure can then be applied to smaller maximum values of z, where the rock is still homogeneous to a good approximation.
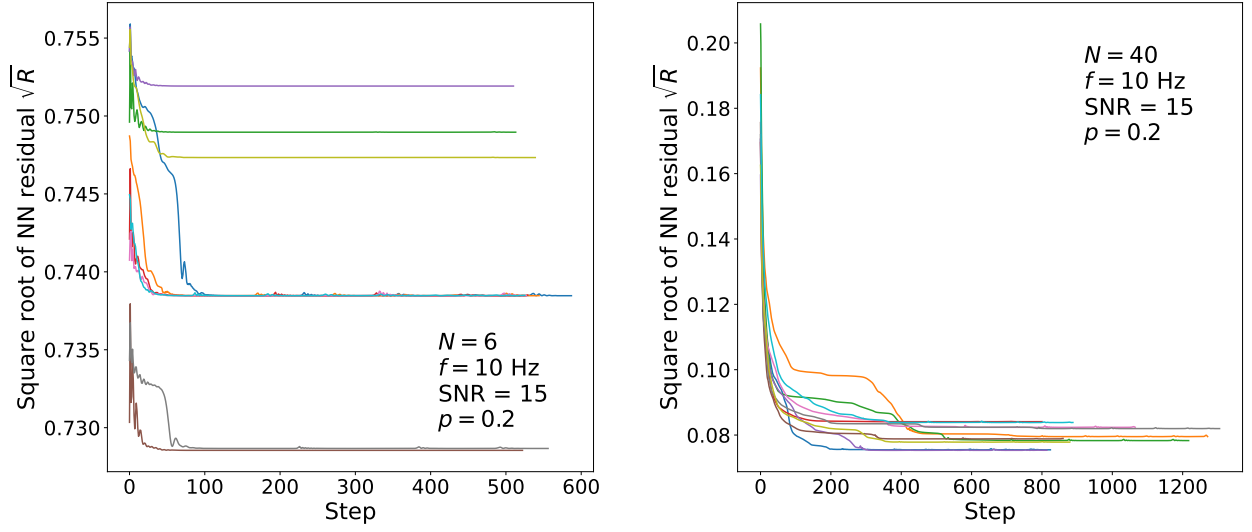
Figure 3: Evolution of the residual during optimization with Adam for $N = 6$ (left) and $N = 40$ (right). The initial positions were pre-optimized with PSO. All curves have parameters $f = 10$ Hz, SNR $= 15$ and $p = 0.2$ and represent the variation from different PSO solutions.
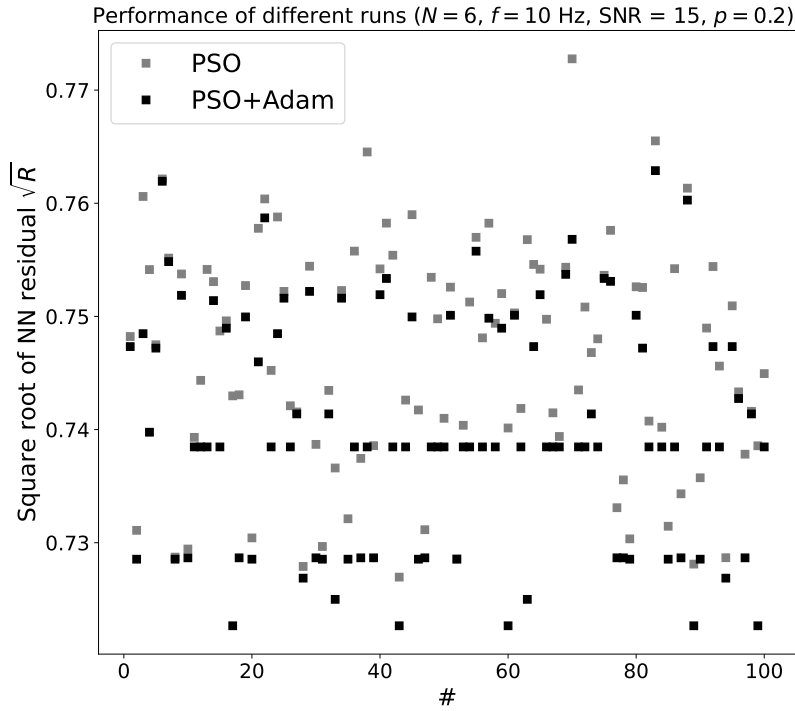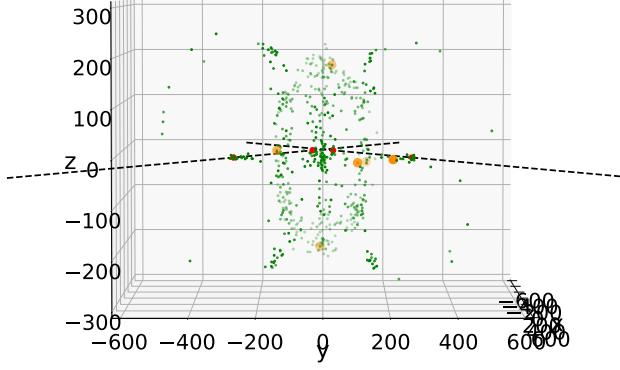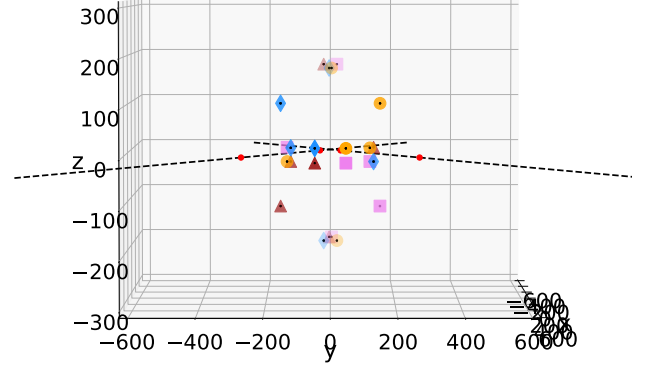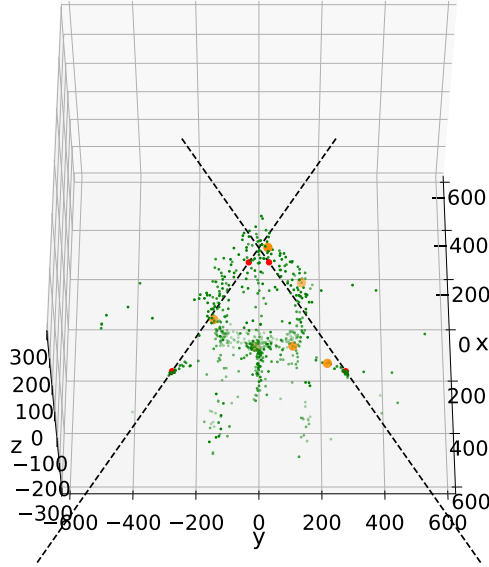

Figure 4: Optimizations with PSO from 100 different runs for $N = 6$, $f = 10$ Hz, $p = 0.2$. The results are shown before (lighter color) and after (darker color) using the PSO results as initial values for Adam.
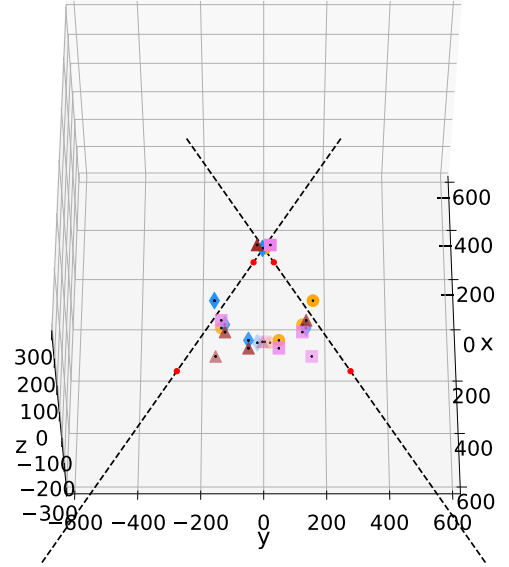
(a) Optimized positions in yz-plane (PSO only)

(b) Optimized positions in yz-plane (PSO+Adam, $\sqrt{R} \approx 0.7384$)

(c) Optimized positions in xy-plane (PSO only)

(d) Optimized positions in xy-plane (PSO+Adam, $\sqrt{R} \approx 0.7384$)

Figure 5: Resulting positions of seismometers in the yz-plane, (a) and (b), and in the xy-plane, (c) and (d), after optimization for $N = 6$, $f = 10$ Hz, SNR $= 15$ and $p = 0.2$. The red dots and black dashed lines represent the mirrors and tunnels of the Einstein Telescope's corner. In (a) and (c), all 100 resulting 6 seismometer positions after optimization with PSO only (light points of Figure 4) are drawn together to visualize the symmetry of the setup. Orange points are the positions from the run with the best residual. In (b) and (d), all runs that have $\sqrt{R} \approx 0.7384$ after optimization with Adam. Differently colored symbols show the four different configurations that lead to the same residual, illustrating the symmetry of the task.

9

by the residual, which results in a penalty for larger values of z. We propose the function

$$R' = R \cdot e^{a \cdot \sum_{i=1}^{N} (z_i/600)^b} \tag{9}$$

with b even, as it exponentially punishes z-values that are too large, while for $|z| < 300$ R' is close to 1. Because of the sum in the exponent, if b is small, configurations with large N will be punished more, while for $e^{\sum_i (z/300)^b}$ with $b \to \infty$ the modified residual is independent of the number or position of any seismometer inside $z = \pm 300$.

For both methods, the residuals are evaluated with the original residual estimate, i.e., Eq. (2), but using the updated z-coordinate. The boundaries for PSO and DE are increased to $\pm 5\,\text{km}$ for the x- and y-coordinate to account for the larger wavelength. The boundaries for the z-coordinate are kept unchanged. These are then used as initial values for the optimization with Adam, either with the clipped z-coordinate or with the constrained cost function from Eq. 9. The resulting z-positions for PSO+Adam and the ratio between the PSO-only residual and the residuals for PSO+Adam for the two methods are shown in Figure 6.
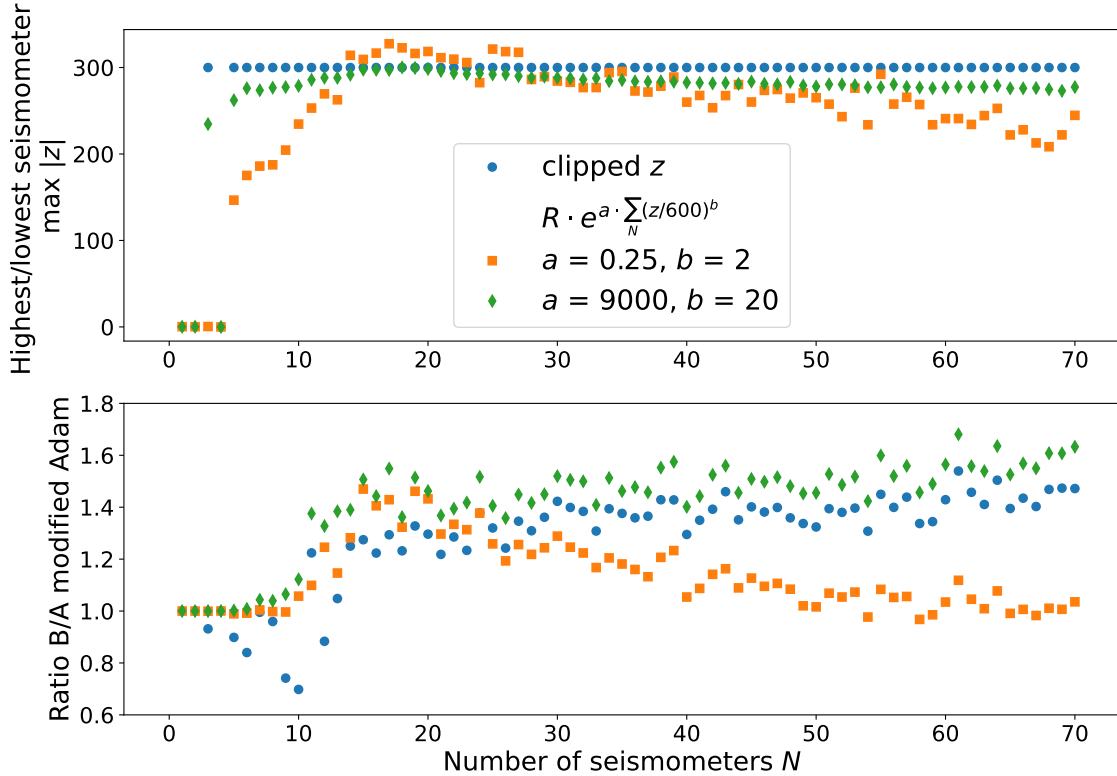

Figure 6: Top: The maximum absolute value of the z-coordinates among all seismometers for different modified optimizations with Adam for $f = 1$ Hz. The initial positions were chosen with PSO. Bottom: The improvement in the residual with respect to the PSO-only results. The PSO-only results are the same as those in Figure 7.

With both methods, we are able to keep the z-positions of the seismometers bounded (upper plot in Figure 7). By construction, clipping leads a flat line at $z = 300$ with a few exceptions at low N, where the optimal seismometer positions are in the plane of ET. The resulting z-coordinates for the modified cost function from Eq. (9) depend on the number of seismometers and the chosen parameters a and b. The curves rise at first up to a maximum at around $N = 15$ and then fall for increasing N. While a is used to steer the maximum allowed value of the z-coordinate, increasing b has the effect of reducing the slope and, therefore, the dependence on N. While for a small number of seismometers, it is not important that the distances of the seismometers from the mirrors in the z-direction are large, for larger values of N, several seismometers would have optimal z-positions of ≥ 300 m, which results in a larger penalty with increasing N in Eq. (9).

Although it is a simple method, PSO+Adam with clipping improves for large values of N compared to the PSO-only optimization result (lower plot in Figure 6). However, this does not work well for some lower values of N. This is not surprising, as the projection of the optimal positions onto $z = 300$ m may result in suboptimal x- and y-positions. For the second method, the improvement depends on the chosen parameters of the function, a and b. In Figure 6, we show results for two extreme choices of b, i.e., $b = 2$ and $b = 20$, where the value

of a is adjusted to exploit as much as possible from the allowed optimization parameter space. The resulting values of a are $a = 0.25$ and $a = 9000$, respectively. The second choice of parameters works better in terms of constraining the seismometers to $|z| < 300$ m and in terms of the achievable residuals. With these values, we were able to reach an improvement of approximately a factor 1.6 for $f = 1$ Hz compared to pure PSO.

Figure 1 shows the optimization results for $f = 1$ Hz for Adam with the modified cost function with $a = 9000$ and $b = 20$ for the different initializations (random, PSO, DE) and compares them to optimizations with PSO and DE alone. Generally, the residuals are larger than for 10 Hz, as the boundary in z-direction hampers the mitigation of Newtonian noise from seismic waves with such large wavelengths. Otherwise, we observe the same trends as before, with the exception of Adam with random initialization, which performs much better than at 10 Hz and finds good positions for values $N > 5$ that are at the same level as the PSO- and DE-initialized optimizations. The reason is probably that the structures in the optimization space are larger due to the larger wavelength, so that random initialization has a much higher probability to find a good minimum. While single optimization runs with DE provide very good residuals already, they can still be improved by approximately 5% with either of the modified Adam optimizations (initialized randomly or with PSO or DE), while Adam initialized randomly or with a single PSO run is faster than single runs with DE by at least the same amount as found for 10 Hz (cf. Figure 2).
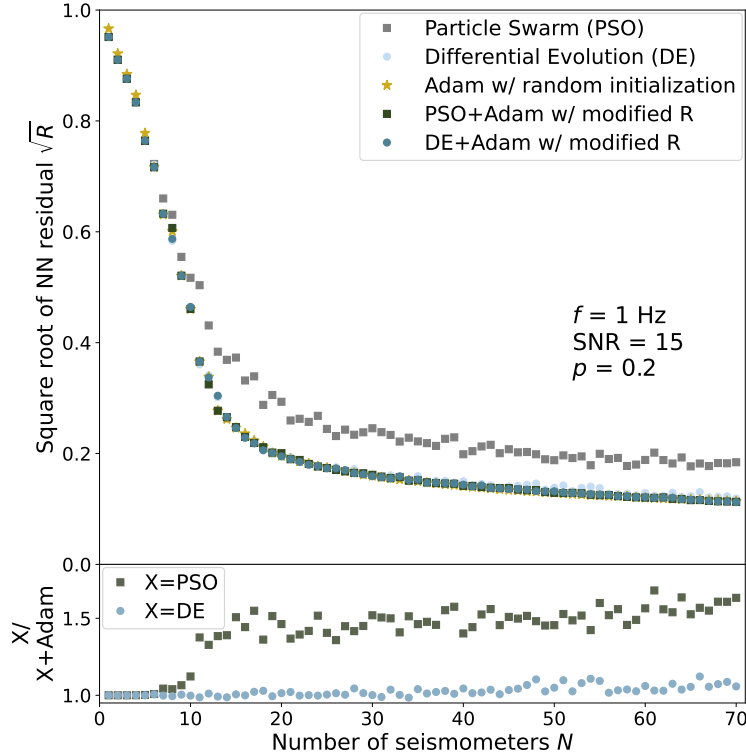

Figure 7: $\sqrt{R}$ for Newtonian noise mitigation as a function of N for $f = 1$ Hz, $p = 0.2$ and SNR $= 15$. The optimization was done with PSO, DE, Adam with random initialization and Adam initialized with PSO and DE, respectively. The optimizations with Adam used the modified cost function from Eq. (9) with $a = 9000$ and $b = 20$. The bottom panel shows improvement in $\sqrt{R}$ obtained when initializing Adam with PSO or DE compared to optimization with PSO or DE alone.

5 Conclusions

We studied the use of gradient-based techniques for optimizing the positions of seismometers at the Einstein Telescope. The seismometers are auxiliary sensors that help to reduce the impact of gravity gradient noise at low gravitational wave frequencies. We reformulated an analytical model for the noise residual based on a Wiener filter in a differentiable way and implemented the model in JAX to make use of its automatic differentiation

capabilities. We then optimized the positions with Adam and compared the results to optimizations with differential evolution and particle swarm optimization. These two metaheuristics had been used in previous studies and were shown to be particularly successful in finding good seismometer positions when repeating the optimization with 100 different initial values. We studied the optimization as a function of the number of seismometers, where we compared the expected noise reduction and the computational costs of the optimization. We chose two reference frequencies, 10 Hz and 1 Hz. For the optimization at 1 Hz, i.e., larger wavelengths, we introduced a constraint term in the optimization to exclude the run-away of seismometer positions outside of the physical volume of the rock.

In general, we find that gradient-based optimization can improve over the optimization with the metaheuristics when initialized with a single optimization run of the metaheuristics. The achievable gains depend on the complexity of the optimization problem. In comparison to the single optimization runs, for $f = 10$ Hz, the achievable improvements are small up to a number of approximately 10 and 25 seismometers for particle swarm optimization and differential evolution, respectively. For larger numbers of seismometers, however, improvements of approximately 125% and 40% can be achieved at moderate additional computational costs, reaching very similar performance when either initialized with differential evolution or particle swarm optimization. Better optima can be found by running particle swarm optimization 100 times with different initial values with the associated larger running time. For large numbers of seismometers, however, the gradient-based optimization still significantly improves over the repeated runs of particle swarm optimization in terms of achievable noise reduction while requiring approximately 50 times less computing time compared to the repeated runs of particle swarm optimization. In addition, we find for the example of an optimization of six seismometer positions that equally good different solutions from the gradient-based optimization reveal the symmetries of the optimization problem. For $f = 1$ Hz, differential evolution as a metaheuristic is able to find seismometer positions that result in only marginally worse residuals than optimizations with Adam, but at a significantly larger computational cost. For this frequency, also randomly initialized optimizations with Adam result in a very good mitigation of Newtonian noise.

In this work, we showed the advantages of gradient-based methods for the optimization of the experimental design of a gravitational wave experiment using the example of the seismometer positions at the Einstein Telescope. While the prospects should be further studied for this particular use case, more applications in the area of gravitational wave experiments may be explored. For the optimization of the seismometer positions, future directions of research include the exploration of other optimizers, in particular second-order optimizers, and the optimization of the positions under more realistic circumstances. These include the use of realistic noise spectra from the drilling campaigns at the candidate sites for the Einstein Telescope, more realistic assumptions on the direction of the seismic waves, on the fraction of P- and S-waves, and on models of the rock, as well as optimizations that are robust with respect to uncertain parts of the seismic noise model.

Acknowledgments

References

[1] B. P. Abbott et al. Observation of Gravitational Waves from a Binary Black Hole Merger. *Physical Review Letters*, 116:061102, 2016. DOI: 10.1103/PhysRevLett.116.061102. arXiv: 1602.03837 [gr-qc].

[2] J. Aasi et al. Advanced LIGO. *Classical and Quantum Gravity*, 32:074001, 2015. DOI: 10.1088/0264-9381/32/7/074001. arXiv: 1411.4547 [gr-qc].

[3] F. Acernese et al. Advanced Virgo: a second-generation interferometric gravitational wave detector. *Classical and Quantum Gravity*, 32:024001, 2015. DOI: 10.1088/0264-9381/32/2/024001. arXiv: 1408.3978 [gr-qc].

[4] B. P. Abbott et al. Multi-messenger Observations of a Binary Neutron Star Merger. *The Astrophysical Journal Letters*, 848:L12, 2017. DOI: 10.3847/2041-8213/aa91c9. arXiv: 1710.05833 [astro-ph.HE].

[5] R. Abbott et al. Observation of Gravitational Waves from Two Neutron Star–Black Hole Coalescences. *The Astrophysical Journal Letters*, 915:L5, 2021. DOI: 10.3847/2041-8213/ac082e. arXiv: 2106.15163 [astro-ph.HE].

[6]   ET Steering Committee Editorial Team. Design Report Update 2020 for the Einstein Telescope, 2020. URL: https://apps.et-gw.eu/tds/?r=18715.

[7]   K. Somiya (for the KAGRA Collaboration). Detector configuration of KAGRA–the Japanese cryogenic gravitational-wave detector. *Classical and Quantum Gravity*, 29:124007, 2012. DOI: 10.1088/0264-9381/29/12/124007. arXiv: 1111.7185 [gr-qc].

[8]   Yoichi Aso et al. Interferometer design of the KAGRA gravitational wave detector. *Physical Review D*, 88:043007, 2013. DOI: 10.1103/PhysRevD.88.043007. arXiv: 1306.6747 [gr-qc].

[9]   P. R. Saulson. Terrestrial gravitational noise on a gravitational wave antenna. *Physical Review D*, 30:732, 1984. DOI: 10.1103/PhysRevD.30.732.

[10] S. Hild et al. Sensitivity studies for third-generation gravitational wave observatories. *Classical and Quantum Gravity*, 28:094013, 2011. DOI: 10.1088/0264-9381/28/9/094013. arXiv: 1012.0908 [gr-qc].

[11] S. Koley et al. Design and implementation of a seismic Newtonian noise cancellation system for the Virgo gravitational-wave detector. *The European Physical Journal Plus*, 139:48, 2024. DOI: 10.1140/epjp/s13360-023-04834-0. arXiv: 2310.17781 [gr-qc].

[12] F. Badaracco and J. Harms. Optimization of seismometer arrays for the cancellation of Newtonian noise from seismic body waves. *Classical and Quantum Gravity*, 36:145006, 2019. DOI: 10.1088/1361-6382/ab28c1. arXiv: 1903.07936 [astro-ph.IM].

[13] F. Badaracco, J. Harms, and L. Rei. Joint optimization of seismometer arrays for the cancellation of Newtonian noise from seismic body waves in the Einstein Telescope. *Classical and Quantum Gravity*, 41:025013, 2024. DOI: 10.1088/1361-6382/ad1715. arXiv: 2310.05709 [astro-ph.IM].

[14] J. Bradbury et al. JAX: composable transformations of Python+NumPy programs, version 0.4.25, 2018. URL: http://github.com/google/jax.

[15] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization, 2014. arXiv: 1412.6980 [cs.LG].

[16] A. G. Baydin et al. Toward Machine Learning Optimization of Experimental Design. *Nuclear Physics News*, 31:25, 2021. DOI: 10.1080/10619127.2021.1881364.

[17] T. Dorigo et al. Toward the end-to-end optimization of particle physics instruments with differentiable programming. *Reviews in Physics*, 10:100085, 2023. DOI: 10.1016/j.revip.2023.100085. arXiv: 2203.13818 [physics.ins-det].

[18] T. Dorigo. Geometry optimization of a muon-electron scattering detector. *Physics Open*, 4:100022, 2020. DOI: 10.1016/j.physo.2020.100022. arXiv: 2002.09973 [physics.ins-det].

[19] S. Shirobokov et al. Black-Box Optimization with Local Generative Surrogates, 2020. arXiv: 2002.04632 [cs.LG].

[20] C. Neubüser, J. Kieseler, and P. Lujan. Optimising longitudinal and lateral calorimeter granularity for software compensation in hadronic showers using deep neural networks. *The European Physical Journal C*, 82:92, 2022. DOI: 10.1140/epjc/s10052-022-10031-7. arXiv: 2101.08150 [physics.ins-det].

[21] M. Aehle et al. Progress in End-to-End Optimization of Detectors for Fundamental Physics with Differentiable Programming, 2023. arXiv: 2310.05673 [physics.ins-det].

[22] M. Aehle et al. Optimization Using Pathwise Algorithmic Derivatives of Electromagnetic Shower Simulations, 2024. arXiv: 2405.07944 [physics.comp-ph].

[23] G. C. Strong et al. TomOpt: differential optimisation for task- and constraint-aware design of particle detectors in the context of muon tomography. *Machine Learning: Science and Technology*, 5:035002, 2024. DOI: 10.1088/2632-2153/ad52e7. arXiv: 2309.14027 [physics.ins-det].

[24] T. Dorigo et al. End-To-End Optimization of the Layout of a Gamma Ray Observatory, 2023. arXiv: 2310.01857 [astro-ph.IM].

[25] N. Wiener. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series: With Engineering Applications.* Wiley, New York, 1949.

[26] G. Cella. Off-Line Subtraction of Seismic Newtonian Noise. In B. Casciaro, D. Fortunato, M. Francaviglia, and A. Masiello, editors, *Recent Developments in General Relativity*, page 495, Milan. Springer, 2000. DOI: 10.1007/978-88-470-2113-6_44.

[27] M. Branchesi et al. Science with the Einstein Telescope: a comparison of different designs. *Journal of Cosmology and Astroparticle Physics*, 07:068, 2023. DOI: 10.1088/1475-7516/2023/07/068. arXiv: 2303.15923 [gr-qc].

[28] F. Badaracco. *Newtonian noise studies in 2nd and 3rd generation gravitational-wave interferometric detectors.* PhD thesis, Gran Sasso Science Institute, 2020.

[29] D. J. Wales and J. P. K. Doye. Global Optimization by Basin-Hopping and the Lowest Energy Structures of Lennard-Jones Clusters Containing up to 110 Atoms. *The Journal of Physical Chemistry*, 101:5111, 1997. DOI: 10.1021/jp970984n. arXiv: cond-mat/9803344 [cond-mat].

[30] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, page 1942, 1995. DOI: 10.1109/ICNN.1995.488968.

[31] R. Storn and K. Price. Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11:341, 1997. DOI: 10.1023/A:1008202821328.

[32] L. J. V. Miranda. PySwarms, a research-toolkit for Particle Swarm Optimization in Python. Version 1.3.0. *Journal of Open Source Software*, 3:433, 2018. DOI: 10.21105/joss.00433.

[33] P. Virtanen et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. Version 1.10.1. *Nature Methods*, 17:261, 2020. DOI: 10.1038/s41592-019-0686-2. arXiv: 1907.10121 [cs.MS].

[34] DeepMind et al. The DeepMind JAX Ecosystem, version 0.2.1, 2020. URL: http://github.com/google-deepmind.