A $O\left(2^{n/2}\right)$ Universal Maximization Algorithm

Research Paper • X-HEC Entrepreneurs 2023-2024

Charles Dana, November 2023

Finite Sum Theorem.

Any real function with n boolean variables can be expressed as a unique finite sum of 2^n real coefficients multiplied by a finite product of boolean variables.

$$\forall f : \{0,1\}^n \to \mathbb{R}$$

$$\forall p \subseteq \{1, \ldots, n\} \, \exists ! a_p \in \mathbb{R}$$

$$\forall x \in \{0,1\}^n$$

$$f(x) = \sum_{p \subseteq \{1,\ldots,n\}} a_p \prod_{i \in p} x_i$$

Finite Sum Theorem.

Proof. Using Lagrange Polynomials and the binary bijection between $\{0,1\}^n$ *and* $\{0,1,2,\ldots,2^n-1\}$*, we show that the indicator* $1_{x=y}$ *for a given* $y \in \{0,1\}^n$*, can be expressed as* $P_y(x)$*, a polynomial evaluation of the natural integer expression of* x*.*

Considering $x = x_1 + 2x_2 + 4x_3 + \ldots + 2^{n-1}x_n$ *and* $x_i^k = x_i$*, it becomes clear that the binomial formula leads to the existence of the formula no matter* f*.*

Unicity comes from contradiction by subtracting two potential candidates, and evaluating the result for a given x *that would highlight a non-zero coefficient.*

Finite Sum Theorem.

What can be said about the real coefficients that compose the sum?

$$\forall p \subseteq \{1,\ldots,n\} : a_p = \sum_{q \subseteq p} (-1)^{|p|-|q|} f\left(\sum_{i \in q} x_i 2^{i-1} \right)$$

Can we derive an estimator for the coefficients without relying on exponential computation? Empirical results suggests that it is possible in most cases, but as we will discuss in this defense, this may not be the most efficient approach for constituting an actionable formula for a significant number of boolean variables.

The relationship with $\mathbb{W}$

the set of real-weighted MAXSAT instances

What is $\mathbb{W}$?

$$\mathbb{W} = \left\{ w = c + \sum_{i=1}^{n} w_i C_i : w_i \geq 0 \right\}$$

$\mathbb{W}$ *is the collection of positively-real weighted clauses on the boolean variables, plus a real constant. We can show a couple of properties on* $\mathbb{W}$*, using the argument:* $\bigvee_{i \in p} x_i + \bigwedge_{i \in p} \neg x_i = 1$

$\{f : \{0,1\}^n \to \mathbb{R}\} \subset \mathbb{W}$*. This result is a consequence of the finite sum theorem, note that there can be multiple candidates* w *to model a real function with* n *boolean variables, depending on the clauses.*

Why using $\mathbb{W}$?

$\mathbb{W}$ is the collection of positively-real weighted clauses on the boolean variables, plus a real constant. Assuming you have constructed an instance $w \in \mathbb{W}$, such that $\forall x \in \{0,1\}^n$, $f(x) = w(x)$, you can use MAXSAT logic to determine $x*$ the optimum of f from the study of w.

MAXSAT is NP-HARD, in practice large instances can only approximate the optimum.

There is no need for constructing the entire instance in order to start deriving an approximation algorithm. The idea is to sample a subset $p \subseteq \{1,\ldots,n\}$ at random. From binomial distribution, the expected cardinal of p is $\frac{n}{2}$.

Overall the algorithm has a complexity $O\left(2^{n/2}\right)$ because each iteration of the while loop relies on the brute-force computation of $2^{|p|}$ coefficients.

Applications.

Maximising Generic Cost Function with n boolean variables

A Framework for an $A*$ on maximization problems

Let $f : \{0,1\}^n \to \mathbb{R}$, be a cost function. According to the Finite Sum Theorem, and by definition of $\mathbb{W}$ we know that there exists a model $w \in \mathbb{W}$ from which we can derive $x*$ such that $f(x*) \geq f(x)$ in $O(2^{n/2})$ complexity.

Although multiple algorithms already exist and are documented on MAXSAT, we present an alternative algorithm that relies on the ability to approximate efficiently a_p through the following formula. Note that because this is an approximation, it can have linear complexity, the tradeoff being between precision and computational efficiency.

$$\forall p \subseteq \{1,\ldots,n\} : a_p \approx \frac{2^{|p|}}{|H(p)|} \sum_{q \in H(p)} (-1)^{|p|-|q|} f\left(\sum_{i \in q} x_i 2^{i-1}\right) : H(p) \subset \mathscr{P}(p)$$

Maximising Generic Cost Function with n boolean variables

A Framework for an $A*$ on maximization problems

Let $\hat{a}_p \approx a_p$ *be the* $O(poly(n))$ *approximation of the exact coefficient* a_p*. One possible approach to solving the model* $w \in \mathbb{W}$ *derived, is to sample* N *coefficients* $\hat{a}_{p_1}, \ldots, \hat{a}_{p_N}$*, ensuring that :*

$\forall i : a_{p_i} \leq 0 \Leftrightarrow \hat{a}_{p_i} \leq 0$ *the sign is preserved.*

$\forall i, j : a_{p_i} \leq a_{p_j} \Leftrightarrow \hat{a}_{p_i} \leq \hat{a}_{p_j}$ *the order relationship is preserved.*

$\forall i, j : |a_{p_i}| \leq |a_{p_j}| \Leftrightarrow |\hat{a}_{p_i}| \leq |\hat{a}_{p_j}|$ *the 'abs' order relationship is preserved.*

We will now assume $|\hat{a}_{p_1}| \geq |\hat{a}_{p_2}| \geq \ldots \geq |\hat{a}_{p_n}|$ *that the coefficients are sorted.*

Maximising Generic Cost Function with n boolean variables

A Framework for an $A*$ on maximization problems

Idea:

$$\hat{a}_{p_1} > 0 \Rightarrow \forall t \in p_1, x_t^* = 1 \textit{ and } \hat{a}_{p_1} \leq 0 \Rightarrow \exists t \in p_1, x_t^* = 0$$

We derive a logic condition from $\hat{a}_{p_1}$

$$\hat{a}_{p_2} > 0 \Rightarrow \forall t \in p_2, x_t^* = 1 \textit{ and } \hat{a}_{p_2} \leq 0 \Rightarrow \exists t \in p_2, x_t^* = 0$$

We derive another logic condition from $\hat{a}_{p_2}$*, if conflicting with an above condition, the above condition is preferred, until the total conflicting contributions of below conditions are greater than the initial one. In our example, if* $|\hat{a}_{p_2}| + |\hat{a}_{p_3}| > |\hat{a}_{p_1}|$ *it can be rational to drop the logic condition derived from* $\hat{a}_{p_1}$.

Maximising Generic Cost Function with n boolean variables

A Framework for an $A*$ on maximization problems

Mitigations:

The sample $p_1, \ldots, p_N$ must be representative of the whole.

$$\forall i, j : |a_{p_i}| \leq |a_{p_j}| \Leftrightarrow |\hat{a}_{p_i}| \leq |\hat{a}_{p_j}| \text{ doesn't imply}$$

$$\forall i, j, k : |a_{p_i}| + |a_{p_j}| > |a_{p_k}| \Leftrightarrow |\hat{a}_{p_i}| + |\hat{a}_{p_j}| > |\hat{a}_{p_k}|$$

Even more so as you keep stacking approximation errors.

There are still 2^n candidates for p, a high entropy function close to $1_{x=x*}$ would not be solvable through this method.

Computing Complex Polynomial Roots

A framework for Universal Polynomial Root Approximation

Let $z \in \mathbb{C}$*, we call* N *the precision parameter.*

We introduce a function $\phi_{\mathbb{C}} : \{0,1\}^{4(N+1)} \to \mathbb{C}$*, that satisfies the property:*

$$|z| < 2^N \Rightarrow \min_{x \in \{0,1\}^{4(N+1)}} |z - \phi_{\mathbb{C}}(x)| \leq 2^{-N}$$

where:

$$\phi_{\mathbb{C}}(x) = (-1)^{x_1} \sum_{k=-N}^{N} x_{N+k+2} 2^k + i(-1)^{x_{2N+3}} \sum_{l=-N}^{N} x_{3N+4+l} 2^l$$

Computing Complex Polynomial Roots

A framework for Universal Polynomial Root Approximation

Let $P : X \mapsto \sum_{k=1}^{n} z_k X^k$ *be a complex polynomial of* $\mathbb{C}[X]$.

$$f : x \mapsto -\,|(P \circ \phi_{\mathbb{C}})(x)|$$

According to the Finite Sum Theorem, under the assumption that there exists a complex root $z^* \in \mathbb{C}$ *to* $P \in \mathbb{C}[X]$ *following* $|z^*| < 2^N$

$$f(x^*) \geq \inf_{\substack{\theta \in [-\pi,\pi] \\ \lambda \in [0,1]}} -\,|P(z^* + \lambda 2^{-N} e^{i\theta})|$$

Such a quantity rapidly converges to 0^- *as* $N \to +\infty$

One way of seeing this is to call $Z = 2^{-N} + \max_{z' : P(z') = 0} |z^* - z'|$ *to obtain the following upper bound to the approximation*

$$|f(x^*)| \leq Z^{n-1} 2^{-N}$$

Practical use for higher degree Polynomials suggest using the alternative logarithmic cost function:

$$f : x \mapsto -\log\left(1 + |(P \circ \phi_{\mathbb{C}})(x)|\right)$$

$$P \sim NP$$

We have an A^* framework for Maximising $f:\{0,1\}^n \to \mathbb{R}$.

Directions.

$$P \stackrel{?}{=} NP$$

Let $d : \{0,1\}^n \rightarrow \{0,1\}$.

I : There exists a SAT instance that models d.

II : The complexity of generating such a SAT Instance is $O(poly(n))$.

III : Real functions with n boolean variables are equivalent to a finite collection of SAT Instances.

$$f \approx (-1)^{d_-} \sum_{k=-N}^{N} d_k 2^k$$

What next?

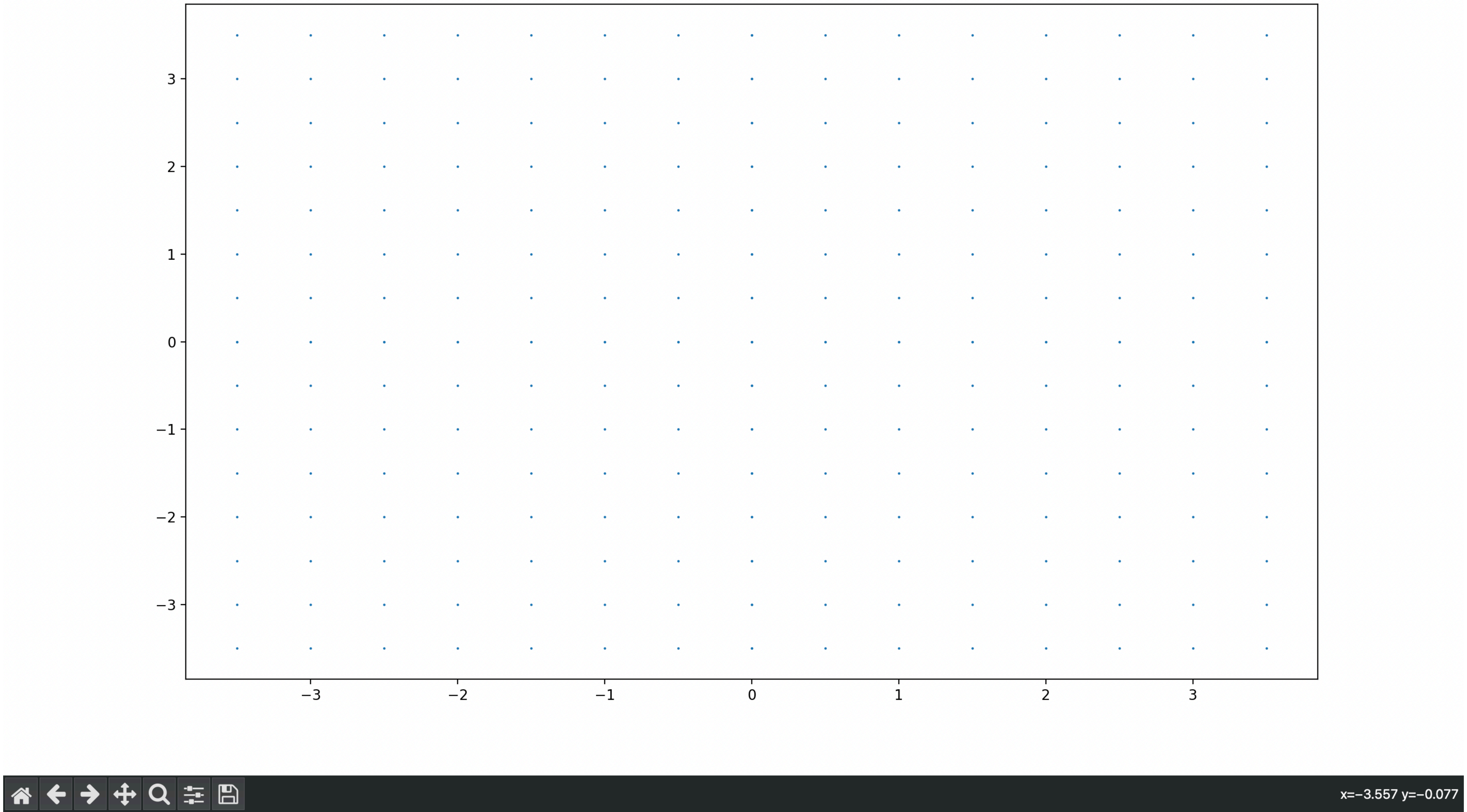
Exploration on the practical Business Implications of the Algorithm

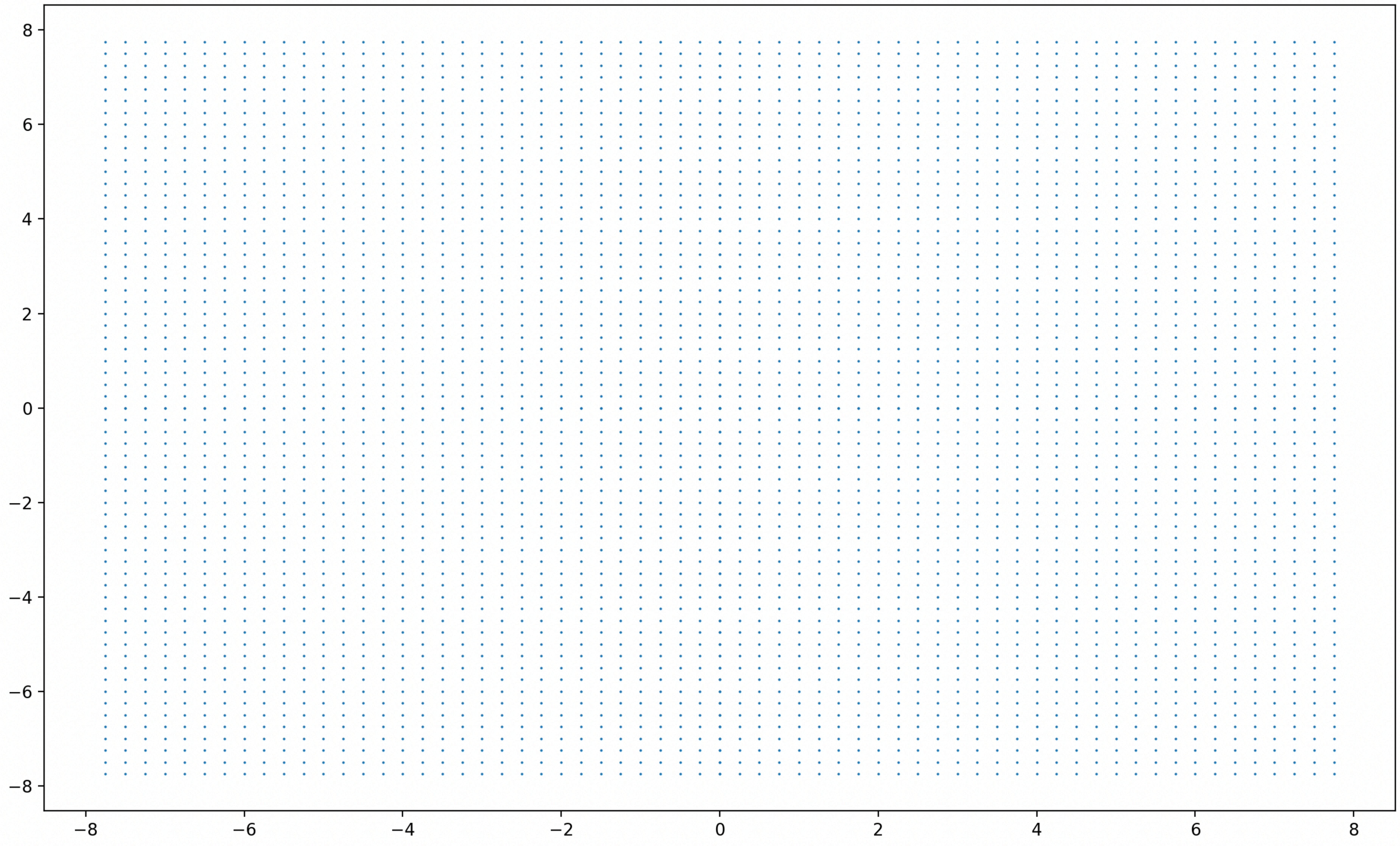
Research & Large-Scale Experiment

EU Funding Application

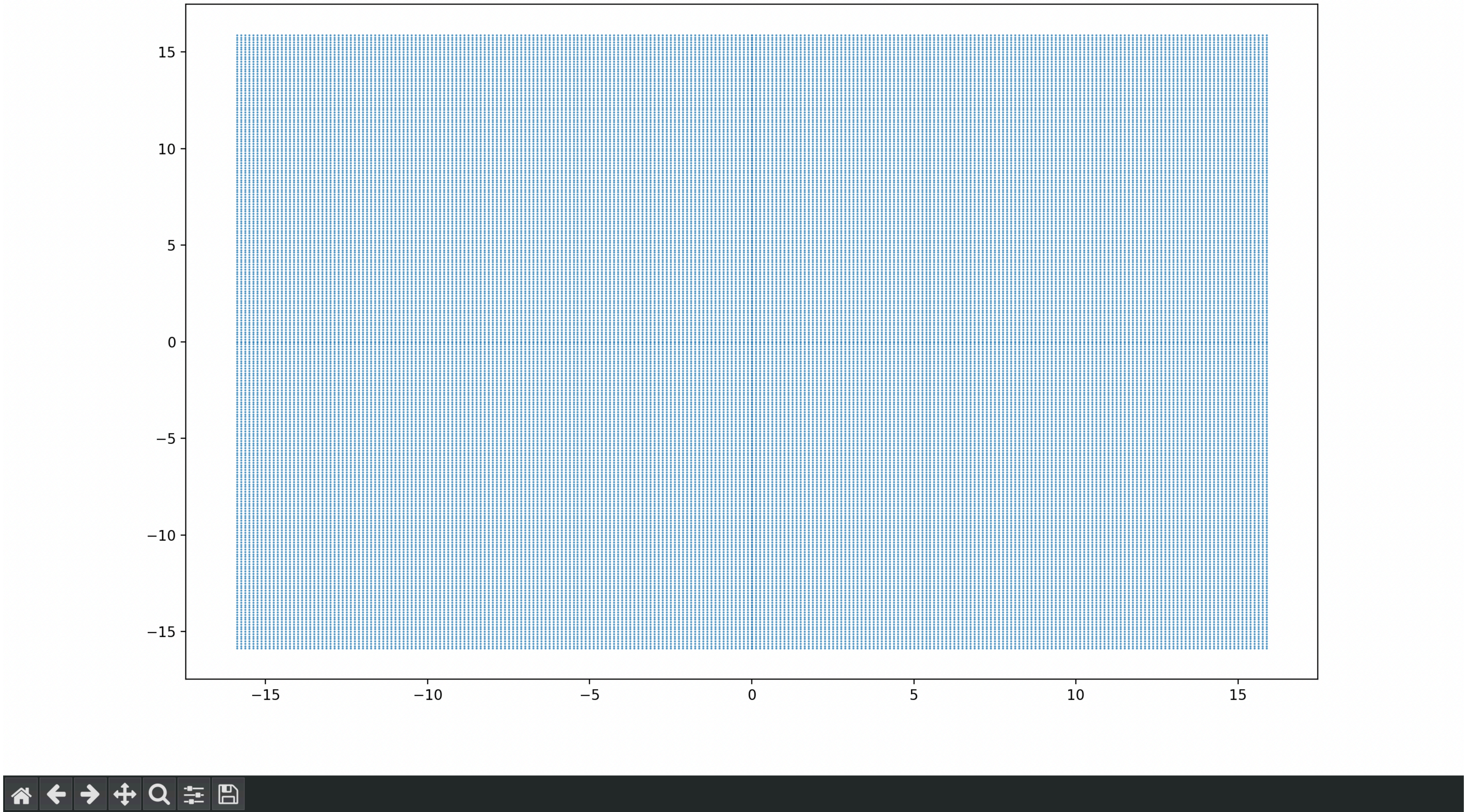
Graphical Intuition for $\phi_{\mathbb{C}}$

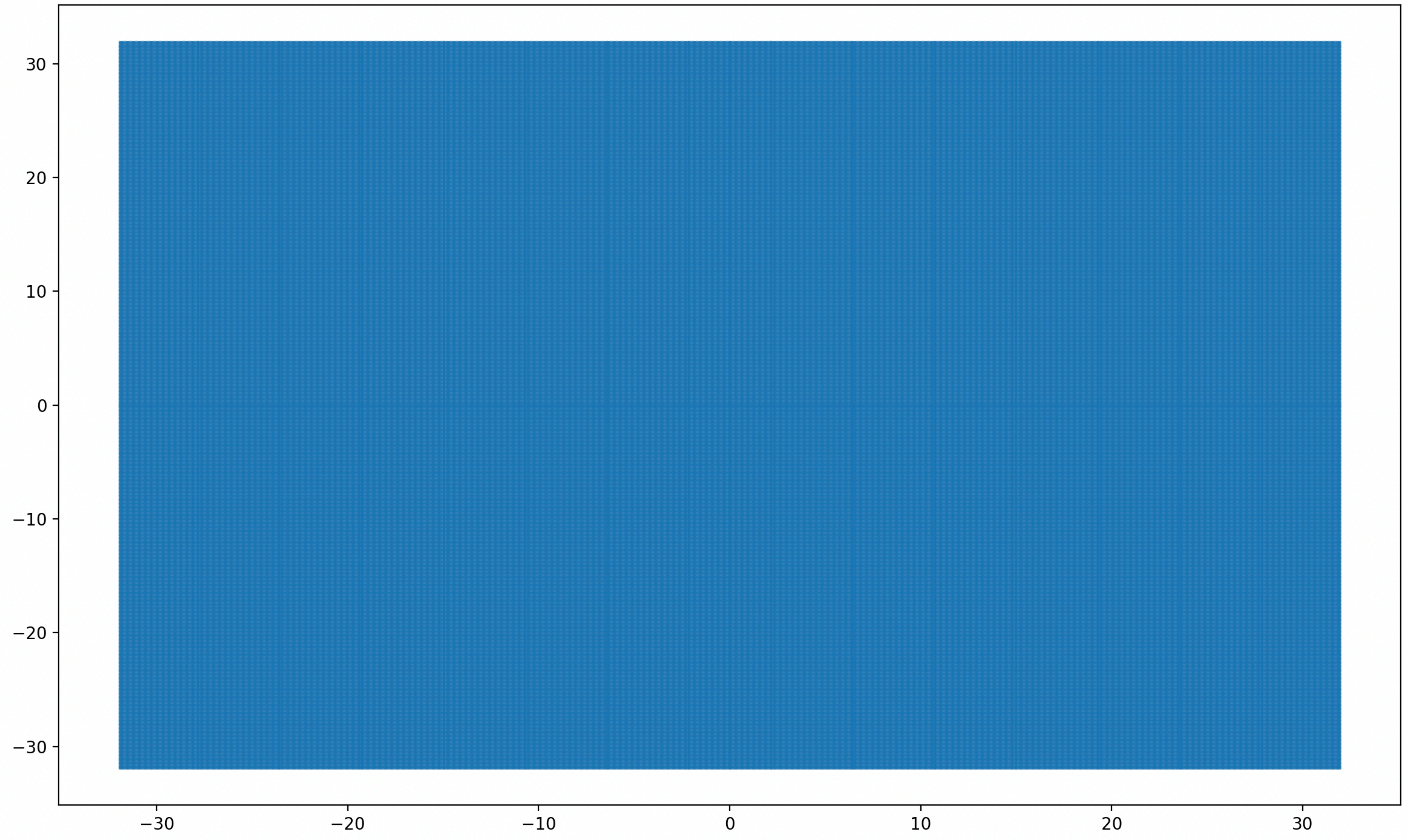
Precision parameter from $N = 1...4$


3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3
x=−3.557 y=−0.077



8
6
4
2
0
−2
−4
−6
−8
−8
−6
−4
−2
0
2
4
6
8


15
10
5
0
−5
−10
−15
−15
−10
−5
0
5
10
15


30
20
10
0
−10
−20
−30
−30
−20
−10
0
10
20
30