The Astrophysical Journal, 979:48 (10pp), 2025 January 20 https://doi.org/10.3847/1538-4357/ad9449

© 2025. The Author(s). Published by the American Astronomical Society.

OPEN ACCESS

# Recovering Pulsar Periodicity from Time-of-arrival Data by Finding the Shortest Vector in a Lattice

Dotan Gazith[1,5], Aaron B. Pearlman[2,3,4,6], and Barak Zackay[1]

[1] Department of Particle Physics and Astrophysics, Weizmann Institute of Science, 76100 Rehovot, Israel; dotan.gazith@weizmann.ac.il, barak.zackay@weizmann.ac.il

[2] Department of Physics, McGill University, 3600 rue University, Montréal, QC H3A 2T8, Canada; aaron.b.pearlman@physics.mcgill.ca

[3] Trottier Space Institute, McGill University, 3550 rue University, Montréal, QC H3A 2A7, Canada

[4] Division of Physics, Mathematics, and Astronomy, California Institute of Technology, Pasadena, CA 91125, USA

*Received 2024 February 12; revised 2024 November 14; accepted 2024 November 15; published 2025 January 16*

## Abstract

The strict periodicity of pulsars is one of the primary ways through which their nature and environment can be studied, and it has also enabled precision tests of general relativity and studies of nanohertz gravitational waves using pulsar timing arrays (PTAs). Identifying such a periodicity from a discrete set of arrival times is a difficult algorithmic problem, In particular when the pulsar is in a binary system. This challenge is especially acute in $\gamma$-ray pulsar astronomy, as there are hundreds of unassociated Fermi-LAT sources that may be produced by $\gamma$-ray emission from unknown pulsars. Recovering their timing solutions will help reveal their properties and may allow them to be added to PTAs. The same issue arises when attempting to recover a strict periodicity for repeating fast radio bursts (FRBs). Such a detection would be a major breakthrough, providing us with the FRB source's age, magnetic field, and binary orbit. The problem of recovering a timing solution from sparse time-of-arrival data is currently unsolvable for pulsars in unknown binary systems, and incredibly hard even for isolated pulsars. In this paper, we frame the timing recovery problem as the problem of finding a short vector in a lattice and obtain the solution using off-the-shelf lattice reduction and sieving techniques. As a proof of concept, we solve PSR J0318+0253, a millisecond $\gamma$-ray pulsar discovered by FAST in a $\gamma$-ray-directed search, in a few CPU minutes. We discuss the assumptions of the standard lattice techniques and quantify their performance and limitations.

*Unified Astronomy Thesaurus concepts:* Pulsars (1306); Astronomy data analysis (1858); Gamma-ray astronomy (628); Computational methods (1965)

## 1. Introduction

The pulsar search problem, recovering the timing parameters of a previously unknown pulsar, is central to pulsar astronomy. Timing pulsars allows us to learn about their ages, magnetic fields, and formation scenarios (D. R. Lorimer & M. Kramer 2004). After obtaining an initial timing solution, precision pulsar timing allows the use of pulsars as tools to study GR (I. H. Stairs 2003), galactic and globular cluster dynamics (B. J. Prager et al. 2017), and the gravitational wave background (J. Antoniadis et al. 2022). When the observations are grouped together and the effective rotational period can be measured on short timescales (for example, radio observations of pulsars), solving the timing problem to phase connect all observations is not computationally demanding (though it is sometimes nontrivial; C. Phillips & S. Ransom 2022).

Periodicity searches become extremely challenging when the data consist of a small set of sparsely spaced times of arrival (TOAs) or phase measurements, as the number of distinct possible timing parameter values rapidly increases with the observation's duration. Although solving this problem has been important for several decades, it has not yet been solved. However, fruitful efforts have been made using semi-coherent techniques, which can, to some extent, also tackle the fully coherent search problem (W. B. Atwood et al. 2006; H. J. Pletsch & C. J. Clark 2014; L. Nieder et al. 2020a, 2020b). As a result, more than a thousand Fermi-LAT unassociated point sources might be pulsars (S. Abdollahi et al. 2022; J. Ballet et al. 2023), where studies based on machine learning have predicted 100–700 of them may be pulsars (S. Germani et al. 2021; J. Coronado-Blázquez 2022; D. V. Malyshev & A. Bhat 2023; K. R. Zhu et al. 2024).

With existing techniques, obtaining a timing solution for a single millisecond pulsar (MSP) for a Fermi-LAT unassociated source is an extremely demanding computational task despite using substantial computational resources. Semi-coherent algorithms are often used, and they require substantial computational resources and compromise on sensitivity (W. B. Atwood et al. 2006; H. J. Pletsch & C. J. Clark 2014; L. Nieder et al. 2020a, 2020b). This scheme is highly suboptimal when applied to solving for MSPs in binary systems (most MSPs are formed through a "recycling" process, where the pulsar's high rotational frequency originates from matter being accreted from a companion in a binary system; D. R. Lorimer 2008).

Extensive distributed volunteer community efforts (e.g., *Einstein@Home*; B. Allen et al. 2013) have been made to bypass this algorithmic difficulty, and many unassociated Fermi-LAT point sources have been blindly followed up using state-of-the-art radio facilities (e.g., D. A. Frail et al. 2018; S. Bruzewski et al. 2023; C. J. Clark et al. 2023). Other techniques, such as cross-matching Fermi-LAT sources with optical sources

[5] Corresponding author.

[6] Banting Fellow, McGill Space Institute (MSI) Fellow, and FRQNT Postdoctoral Fellow.

Original content from this work may be used under the terms of the Creative Commons Attribution 4.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

exhibiting periodic modulation, have enabled the recovery of several of the timing parameters (e.g., precise position, proper motion, and orbital period), reducing the computational load by more than 8 orders of magnitude and making the recovery effort feasible with current algorithms (L. Nieder et al. 2020b).

Another outstanding example of the need for an algorithmic solution to the pulsar search problem is the effort to search for a strict periodicity in the arrival times of repeating fast radio bursts (FRBs). FRBs are a class of extragalactic astrophysical sources, characterized by extremely luminous radio bursts (J. M. Cordes & S. Chatterjee 2019; E. Petroff et al. 2019; M. Bailes 2022), with durations ranging from nanoseconds to milliseconds (W. A. Majid et al. 2021; K. Nimmo et al. 2022; M. P. Snelders et al. 2023). These radio bursts have a wide range of applications, including cosmological studies (e.g., see A. Walters et al. 2018; J. P. Macquart et al. 2020), understanding the FRB emission engine (e.g., see The CHIME/FRB Collaboration et al. 2020; A. B. Pearlman et al. 2024), and distinguishing between different source types (e.g., see F. Kirsten et al. 2022; M. Bhardwaj et al. 2024). Some FRBs have been observed to emit multiple bursts and are referred to as repeating FRBs (e.g., see The CHIME/FRB Collaboration et al. 2019; E. Fonseca et al. 2020; The CHIME/FRB Collaboration et al. 2023). It is still unclear if all FRBs repeat and what the burst emission statistics are (J. M. Cordes & S. Chatterjee 2019).

An important hint as to why this search is expected to be computationally hard is the discovery of many-day periodicity in the activity of some repeating FRBs, hinting at a binary origin (The CHIME/FRB Collaboration et al. 2020). The discovery of a subsecond periodicity in the so-far nonrepeating FRB 20191221A suggests that some FRB sources may be powered by rotating neutron stars with periodic radio emission (The CHIME/FRB Collaboration et al. 2022). However, a timing solution from a repeating FRB still remains elusive, despite substantial search efforts and the detection of hundreds of bursts from several sources (D. Li et al. 2021; J.-R. Niu et al. 2022; H. Xu et al. 2022; C. Du et al. 2024).

A leading candidate for the FRB engine is a highly magnetized neutron star (E. Platts et al. 2019), based on the short duration of the observed radio emission and the similar phenomenological characteristics shared with pulsars (A. B. Pearlman et al. 2018). Nearly all phenomena (including giant pulses) related to neutron stars have temporal properties that reveal the neutron star rotation (M. B. Mickaliger et al. 2012). If repeating FRBs also share this property, it is, in principle, possible to to obtain a timing model using the arrival times of the radio bursts, where the arrival times of the bursts would cluster in rotational phase, similar to giant pulses from the Crab pulsar (M. B. Mickaliger et al. 2012).

Pulsar timing models can be incredibly precise and sensitive to many significant digits in the rotation frequency, frequency derivatives, sky position, and Keplerian (and post-Keplerian) orbital parameters. Measuring all the above parameters for a repeating FRB would allow us to study their astrophysical formation scenario through measurements of their age, surface magnetic field, orbital period, eccentricity, and binary mass function. Since FRBs are very extreme systems, orders of magnitude brighter than other known Galactic pulsars (J. M. Cordes & S. Chatterjee 2019), their formation scenario may reveal rare, yet important, phenomena related to the formation of compact objects and perhaps even their influence on their surroundings.

### 1.1. The Pulsar Search Problem and Existing Solutions

The observed arrival times of the bursts, $t_{\rm obs}$, can be modeled as

$$t_{\rm obs} = t_{\rm em} + \Delta t_{\rm orb} + \Delta t_{\rm prop}, \tag{1}$$

where $t_{\rm em}$ is the emission time of the burst, $\Delta t_{\rm orb}$ is the delay due to the orbital motion of the source, and $\Delta t_{\rm prop}$ is the delay due to the propagation in the solar system (geometric and relativistic corrections). A perfectly periodic source satisfies

$$t_{\rm em} \bmod P_{\rm rot} = \phi_t. \tag{2}$$

Equivalently, we can write

$$f_{\rm rot}\, t_{\rm em} = K + \phi, \tag{3}$$

where $f_{\rm rot}$ is the source's rotational frequency and $K$ is an integer. A source whose rotation rate is changing linearly with time satisfies

$$f_{\rm rot}\, t_{\rm em} + \dot{f}_{\rm rot} \frac{t_{\rm em}^2}{2} = K + \phi. \tag{4}$$

To characterize the computational hardness of finding the timing solution, we can estimate the number of "independent" timing models that we would need to enumerate in a brute-force search for the timing model by

$$\Lambda \equiv N_{\rm rot} N_{\rm geom} N_{\rm orb}, \tag{5}$$

where $N_{\rm rot} \equiv N_f \times N_{\dot{f}}$, and $N_f \equiv \frac{f}{\delta_f}$ and $N_{\dot{f}} \equiv \frac{\dot{f}}{\delta_{\dot{f}}}$, where $\delta_f$ and $\delta_{\dot{f}}$ are the typical measurement errors in the timing model. Similarly (but perhaps with more complications), $N_{\rm geom}$ is the characteristic number of options for "independent" sky positions (and proper motion and parallax), and $N_{\rm orb}$ is the number of independent orbital configurations.

When trying to find a timing solution for an MSP using Fermi-LAT data, the typical numbers are as follows:

1. Hundreds of arrival times, spread over 15 yr, with significant association probabilities ($\gtrsim$0.2).
2. The pulsar's spin frequency and its derivatives are unknown ($N_{\rm rot} \in [10^{13}, 10^{18}]$).
3. The pulsar's position is known only to ~0.1°. The precision required for a phase-connected timing solution is $\sim 10^{-3}$–1 arcseconds, depending on the rotation frequency and duty cycle. In some cases, proper motion and parallax may also be required, yielding $N_{\rm geom} \in [10^6, 10^{12}]$ options.
4. The binary orbit is unknown (i.e., there are five missing Keplerian parameters), resulting in $N_{\rm orb} \in [10^{10}, 10^{20}]$ different options.

Similar numbers are encountered when searching for periodicities from repeating FRBs. The "brute-force" method for searching a timing model is to take a group of bursts, try all combinations of parameters, compute for all arrival times their corresponding rotational phase, and perform a statistical test to detect deviations from a uniform distribution. For short observation durations (minutes–hours), such an enumeration is feasible, and indeed, for RRATs (a special class of neutron stars, emitting pulses irregularly), this can successfully produce short-duration timing models that are phase-connected between observations using heuristic software and manual procedures (M. A. McLaughlin et al. 2009). However, this approach does not scale for long observing durations (several years) and/or when including a binary orbit (with a binary period shorter than the observation duration). The number of options required for a complete enumeration easily exceeds $10^{30}$ for recovering a timing solution involving a binary orbit. Since this is unfeasible for the foreseeable future, any viable path includes an algorithmic method that is drastically different from brute-force enumeration.

The current state-of-the-art algorithms used for solving the pulsar search problem are the semi-coherent enumeration algorithms (H. J. Pletsch & C. J. Clark 2014; L. Nieder et al. 2020a). These algorithms utilize a special detection statistic that reduces the number of trials by reducing the coherence time, trading off the overall search sensitivity for a much reduced computational complexity. Currently, these algorithms use one of the largest computing networks on the planet, *Einstein@Home* (B. Allen et al. 2013), which spends up to 1000 core years per target. Even with the best computing resources, using state-of-the-art methods, the pulsar search problem could be solved blindly only when restricted to isolated pulsars and with reduced sensitivity (due to a relatively short coherence time used to reduce the computational load).

### 1.2. Our Contributions

In a series of papers, we cast the pulsar search problem into the problem of finding a short vector in a lattice and show the remarkable utility of lattice reduction and sieving for solving the pulsar search problem. A proof of concept for an algebraic algorithm that solves the pulsar search problem (in contrast to the enumeration techniques currently employed) is presented in this paper. The algorithm exactly converts the astrophysical question into the problem of finding the shortest (nontrivial) vector in a lattice. This class of algorithms has been extensively developed with cryptanalysis applications in mind. As far as we know, this is the first application of these algorithms, in high dimension and of cryptographic hardness, outside of cryptography and number theory, although it was used as a speed-up tool in satellite navigation systems (P. J. Teunissen 1993).

In this paper, we first show how the pulsar detection problem on sparse data could be written as finding a short vector in a lattice. This problem, although NP-hard (S. Khot 2004), is surprisingly solvable for lattices with very high dimension, due to there being a large body of algorithms, such as the lattice reduction algorithms, LLL (A. K. Lenstra et al. 1982), BKZ (C. Schnorr & M. Euchner 1994; N. Gama et al. 2010), and Gaussian Sieve algorithms (P. Q. Nguyen & T. Vidick 2008; A. Becker et al. 2015), with the most advanced algorithms combining both concepts (L. Ducas 2018; M. R. Albrecht et al. 2019).

We then find the shortest vector in the resulting lattices using state-of-the-art off-the-shelf shortest vector problem (SVP) solvers (M. R. Albrecht et al. 2019; The FPLLL development team 2024). We demonstrate the algorithm's applicability to realistic situations using simulations and gamma-ray photon arrival times from Fermi-LAT. Empirically and heuristically, we describe the conditions for the algorithm's success.

### 1.3. Review of Existing Solutions to the Shortest Vector Problem

A lattice $\mathcal{L}$ is the set of all linear combinations with integer coefficients of a basis of vectors (row vectors of matrix $L$, where the vectors may contain any real values):

$$\boldsymbol{v} \in \mathcal{L} \Longleftrightarrow \exists\ \boldsymbol{x} \in \mathbf{Z}^n \text{ s.t. } \boldsymbol{v} = \boldsymbol{x}L, \tag{6}$$

where $n$ is the number of basis vectors and is called the lattice dimension. The SVP refers to the following problem:

$$\min_{\vec{0} \neq \boldsymbol{v} \in \mathcal{L}} ||\boldsymbol{v}||^2. \tag{7}$$

Lattice Sieving methods generate a set $V$ of many vectors of typical length $l$, and then pairs of vectors ($\boldsymbol{v}_1, \boldsymbol{v}_2 \in V$) are iteratively used to compute $\boldsymbol{w} = \boldsymbol{v}_1 - \boldsymbol{v}_2$. If $\boldsymbol{w}$ is shorter than either $\boldsymbol{v}_1$ or $\boldsymbol{v}_2$, then it replaces them in $V$. This process is done iteratively until convergence. If the set $V$ has more than $\sim 2^{0.21n}$ vectors,[7] then the typical lengths of vectors in the set $V$ shrink more and more until the shortest vector in the lattice is found.

Lattice reduction techniques are seeking a factorization of the lattice

$$L^t = QRU, \tag{8}$$

such that $Q$ is orthonormal, $U$ is unimodular (a matrix with determinant 1 and integer coefficients), and $R$ is upper triangular. The reduction process gradually updates $U$ by finding ways to make the matrix $R$ more favorable for enumeration algorithms that seek to solve $\boldsymbol{R}\boldsymbol{x} = \boldsymbol{v}_{\text{target}}$, such as Babai's nearest plane algorithm (L. Babai 1986) whose error is proportional to $\sum_i |R_{ii}|^2$. This is accomplished by maximizing the bottom values on the diagonal of $R$. Reduction algorithms usually achieve a matrix, $R$, with

$$\frac{R(i, i)}{R(j, j)} = \delta_b^{j-i}, \tag{9}$$

where $b$ is the block size of BKZ and $\delta_b \approx \left(\frac{b}{2\pi e}\right)^{1/b}$. The complexity of BKZ is superexponential in $b$. The larger the $b$, the more convenient the enumeration. This can be intuitively understood as meaning that the greater $b$ is, the closer the values in $R$'s diagonal are, which leads to easier enumeration, but this reduction is harder to achieve. The greatest advantage of lattice reduction techniques is their ability to find very short vectors,[8] if such exist, even if the dimension is large.

The current state-of-the-art algorithm for solving the SVP (M. R. Albrecht et al. 2019) combines both approaches, maintaining a database of short vectors on gradually increasing sublattices (using Babai's nearest plane algorithm (L. Babai 1986), utilizing a reduced basis), repeatedly keeping them short (using a sieve), and exploiting the database's size for an eventual sieve to solve a few extra dimensions for free (L. Ducas 2018).

## 2. Pulsar Detection as Short Vector in a Lattice—The Basic Construction

We first introduce the basic timing model used in our framework:

$$t_i = (K_i + \phi + \epsilon_i)P, \tag{10}$$

[7] The exact limit is $\left(\frac{4}{3}\right)^{n/2}$, as proven by P. Q. Nguyen & T. Vidick (2008).

[8] Very short as compared to the expected shortest vector in a random lattice, with each reduction algorithm supplying some guarantees that are typically surpassed in practice.

where $t_i$ is the $i$th TOA, $P$ is the pulsar's rotational period, $K_i$ is an integer number of pulsar rotations since the $t = 0$ reference time, and $\epsilon_i$ is the pulse phase at which this photon arrived (in the interval [−0.5, 0.5]). Assuming a Gaussian pulse profile with standard deviation $\sigma$ (where $\sigma$ is proportional to the "duty cycle"), $\epsilon_i \sim N(0, \sigma)$. By rearranging Equation (10), we obtain

$$ft_i - K_i - \phi = \epsilon_i, \tag{11}$$

where $f = 1/P$ is the rotational frequency of the pulsar. Since we assume the pulse profile is of Gaussian shape, according to the Neyman–Pearson lemma (J. Neyman & E. S. Pearson 1933), the strongest statistical test to decide between the null hypothesis, uniform phase residuals, and the alternative Gaussian pulse profile is

$$\mathcal{T} = \sum_i \epsilon_i^2. \tag{12}$$

For specific values of $f$, $\phi$, and $K_i$ to indicate a potential detection, they should first be better than the alternative values. So, even before considering the significance, we need to find the best values for the parameters, which requires minimizing $\mathcal{T}$. To prevent undesired solutions of extremely high frequencies (for example, due to the clock frequency of the detector), we also enforce a Gaussian prior on the parameters:

$$V\left[\frac{f}{f_{\text{prior}}}\right] = 1,$$
$$V[\phi] = 1. \tag{13}$$

When adding them to the test statistic, we will ensure each of those priors contributes $\sigma^2$, the same as any other coordinate:

$$\mathcal{T} = \sum_i \epsilon_i^2 + \left(\sigma\frac{f}{f_{\text{prior}}}\right)^2 + (\sigma\phi)^2. \tag{14}$$

The lattice structure arises from the restriction that the $K_i$ values are integers, while $f$ and $\phi$ are unknown and of some precision. The equation needs to be strictly linear in the unknowns and all unknown coefficients must be strictly integers, as required by the lattice solver that we use (G6K; M. R. Albrecht et al. 2019). For this purpose, Equation (11) can be re-expressed as

$$\left[\frac{f}{d_f}\right] d_f t_i - K_i - \left[\frac{\phi}{d_\phi}\right] d_\phi = \epsilon_i. \tag{15}$$

Here, $d_f$ is the measurement resolution of $f$ in the integer solution, which we choose following

$$d_f \ll \frac{\sigma}{(\max_i t_i - \min_i t_i)}, \tag{16}$$

to ensure the rounding resolution is much smaller than the expected parameter's precision in order to avoid rounding errors. Writing the lattice basis vectors as rows of the following matrix, we can write:[9]

$$L_{\text{per}} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & & \dots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 & 0 \\ d_f t_0 & d_f t_1 & d_f t_2 & \dots & d_f t_n & \eta_t & 0 \\ d_\phi & d_\phi & d_\phi & \dots & d_\phi & 0 & \eta_\phi \end{pmatrix}, \tag{17}$$

where we introduced $\eta_t$, $\eta_\phi$ to account for the priors in Equation (14), and a choice for them that will give the same expected loss in every coordinate is

$$\eta_t = \sigma\frac{d_f}{f_{\text{prior}}}, \quad \eta_\phi = \sigma d_\phi. \tag{18}$$

Using this lattice basis definition, the vectors in the lattice will have the components of our test statistic from Equation (14) as their entries, and its length that we minimize when searching for the shortest vector corresponds to the test statistic, incorporating the phase residuals and priors.

## 3. Extending the Lattice-based Solution to Other Unknowns

We have described the technique for using the lattice framework to find simple, perfectly periodic solutions. However, the full advantage of using the lattice approach is incorporating many other timing parameters into the model without significantly increasing the computational complexity. The most prominent of these are the pulsar spin-down parameters ($\dot{f}$, $\ddot{f}$, ...), the barycentric-correction parameters, and some of the Keplerian orbit parameters.

### 3.1. Spin-down Parameters

The spin frequency of rotating neutron stars can change due to processes such as magnetic breaking and accretion. We can calculate the rotational phase of each TOA, $t_i$, using

$$\phi + K_i + \varepsilon_i = \int_{t_{\text{ref}}}^{t_i} f(t)dt, \tag{19}$$

where $K_i$ are integers, $t_{\text{ref}}$ is the reference time, and $f(t)$ is the instantaneous rotation frequency. Expanding $f(t)$ as a Taylor series and integrating, the arrival times satisfy the following equation:

$$K_j + \epsilon_j = \phi + ft_j + \sum_{k=2}^{m} \frac{f^{(k-1)}}{k!} t_j^k. \tag{20}$$

Similarly to Equation (15), we can equivalently write

$$K_j + \epsilon_j = \phi + \left[\frac{f}{d_f}\right] d_f t_j + \sum_{k=2}^{m} \left[\frac{f^{(k-1)}}{d_{f^{(k)}} k!}\right] d_{f^{(k-1)}} t^k. \tag{21}$$

[9] The G6K lattice solver (M. R. Albrecht et al. 2019) also requires that the input matrix's entries are integers, so we multiply the elements by a large number and round. This number is chosen to be much larger than the largest expected solution coefficient divided by the phase residual's resolution, typically $\gg \frac{f_{\text{prior}}}{d_f \sigma^2}$.

This form of the equation can be naturally encapsulated by adding a few additional vectors to the lattice:

$$L_{\text{spin-down}} = \begin{pmatrix} I_{n\times n} & \mathbb{O}_{n\times(m+1)} \\ S_{(m+1)\times n} & \eta_{(m+1)\times(m+1)} \end{pmatrix}, \tag{22}$$

where $I_{n\times n}$ is the identity matrix, $0_{n\times(m+1)}$ is a zero matrix, $S_{(m+1)\times n}$ is a matrix containing the spin-down vectors, and $\eta_{(m+1)\times(m+1)}$ is a diagonal matrix. $S_{(m+1)\times n}$ has the form

$$S_{(m+1)\times n} = \begin{pmatrix} d_\phi & \dots & d_\phi \\ d_f t_0 & \dots & d_f t_{n-1} \\ \frac{1}{2} d_{f^{(1)}}(t_0)^2 & \dots & \frac{1}{2} d_{f^{(1)}}(t_{n-1})^2 \\ \vdots & \ddots & \vdots \\ \frac{1}{m!} d_{f^{(m-1)}}(t_0)^m & \dots & \frac{1}{m!} d_{f^{(m-1)}}(t_{n-1})^m \end{pmatrix}, \tag{23}$$

and the elements of $\eta_{(m+1)\times(m+1)}$ are given by

$$\eta(1, 1) = \frac{\sigma}{d_\phi}, \tag{24}$$

$$\eta(k+1, k+1) = \sigma \frac{d_{f^{(k)}}}{f^{(k)}_{\max}}. \tag{25}$$

### 3.2. Accounting for an Unknown Position

The Fermi satellite follows Earth's orbital motion, which, if uncorrected, will leave a 500-second differential residual in the photon arrival times. The main component of the time correction is to project the satellite's position vector relative to the solar system's barycenter along the direction of the source.[10] Since the required time resolution for efficient MSP recovery is on the order of $10^{-4}$ s, the sky position needs to be known to a precision that is better than $\frac{10^{-4}}{500}$ rad $\approx$10 mas. The localization precision of sources in the 4FGL catalog is roughly 0.1°. Thus, for pulsar searches, there are $10^9$ different trial positions that are possible in a blind search.

Moreover, the proper motion of the pulsar can also substantially affect the timing solution (especially for MSPs) because the resolution required by the proper motion is on the order of $1\frac{\text{mas}}{\text{yr}}$. A pulsar that is located at a distance of 1 kpc, with a tangential velocity of $100\,\text{km}\,\text{s}^{-1}$, will have a proper motion on the order of $10\,\text{mas}\,\text{yr}^{-1}$, which will introduce substantial timing residuals. To solve for the pulsar's precise position with the lattice, we notice that the space of all possible corrections is linear (although the coordinate transformation between this space and the commonly used coordinates is not completely linear). We collectively denote all of the positional parameters as $\boldsymbol{\psi}$, and their collective phase delay (time delay times frequency) as $F(\boldsymbol{\psi})$. We can then write

$$\begin{aligned} ft_j &= K_j + F(t_j, \boldsymbol{\psi}) \\ &= K_j + F(t_j, \boldsymbol{\psi_0}) + (\boldsymbol{\psi} - \boldsymbol{\psi}_0)\frac{dF}{d\boldsymbol{\psi}}(t_j, \boldsymbol{\psi_0}). \end{aligned} \tag{26}$$

This linearized model for the timing solution could be added to the lattice via the same procedure described for spin-down parameters. In general, this approximation corresponds to trying to reconstruct a three-parameter function with nonlinear constraint (a unit vector pointing at the pulsar's direction) using two unconstrained parameters. We might naively expect the approximation error to grow as

$$\left| (\boldsymbol{\psi} - \boldsymbol{\psi}_0)_l \frac{d^2F}{d\psi_l d\psi_m}\Big|_{(t_j, \boldsymbol{\psi_0})} (\boldsymbol{\psi}' - \boldsymbol{\psi}_0)_m \right| \tag{27}$$

and result in an intolerable error (barycentric corrections that are nonphysical) for relatively small position offsets. However, the Fermi satellite's motion around the solar system's barycenter is nearly planar, and if we look in ecliptic coordinates, we need only two weakly constrained parameters to linearly combine $(x_{\text{Fermi}}(t_i), y_{\text{Fermi}}(t_i))$. This means that even large offsets will still yield a barycentric correction corresponding to some pulsar position. Also, the mixing between the barycentric correction and the first frequency derivative is negligible for reasonable search parameters.[11] Therefore, there is no need for external enumeration of the exact position of the pulsar, and our linear approximation is sufficient.

### 3.3. Adding a Circular Orbit into the Lattice

Many of the pulsars that we aim to detect (for example, most MSPs) are in binary systems. Therefore, we construct a feasible algorithm for blind detection for such systems.

This issue was not addressed in the previous blind pulsar search surveys because brute-force enumeration of all of the orbital parameters, along with the spin frequency, spin frequency derivatives, sky position, and proper motion, is unfeasible. Brute-force enumeration of all of these parameters can easily accumulate to more than $10^{30}$ independent options in the parameter space.

In Section 7, we demonstrate that, with the lattice-based solution, we can solve for position and spin-down in ∼100 core seconds, which, even with the simple, brute-force approach, calls us to reconsider the binary search problem.

The lattice-based solution efficiently solves linear integer least-squares problems. Therefore, to solve for the orbit, we must also linearize the phase space of all circular orbits as much as possible. The orbital reference phase and semimajor axis are trivially linearizable:

$$\begin{aligned} v_{1,\text{orb}} &= \sin(\Omega_{\text{orb}} t) \\ v_{2,\text{orb}} &= \cos(\Omega_{\text{orb}} t). \end{aligned} \tag{28}$$

The orbital frequency, $\Omega_{\text{orb}}$, is more challenging, as periods that differ by an integer number of orbits during the observation duration are approximately orthogonal. Therefore, we must divide the parameter space into a union of many different linear spaces, covering all of the options for $\Omega_{\text{orb}}$. For a circular orbit with an orbital period of 10 hours, a semimajor axis of 1 light second, an observation duration of 10 years, and a target timing precision of 0.1 ms, this amounts to $\frac{a}{\sigma P}\frac{T}{P_{\text{orb}}} \approx 10^8$ different period trials. Since each trial requires solving the SVP problem (at least 1–100 CPU seconds, depending on dimensions; see L. Ducas et al. 2021), this will be extremely demanding, with

[10] There are also general relativistic propagation effects and motions due to other planetary bodies in the solar system that need to be corrected. This can be accounted for, e.g., using the barycentering routines available in the PINT software package (J. Luo et al. 2021).

[11] $\varepsilon = \frac{1}{2}\dot{f}\left(2T\Delta\alpha\frac{AU}{c}\right) \approx 3\cdot 10^{-3}\left(\frac{\dot{f}}{-10^{-11}\text{s}^{-2}}\right)\left(\frac{\Delta\alpha}{0.1°}\right).$

an estimated cost of hundreds of core years. We can reduce this number of trials by adding the following vectors to the lattice:

$$v_{3,\mathrm{orb}} = t\,\sin(\Omega_{\mathrm{orb}} t)$$
$$v_{4,\mathrm{orb}} = t\,\cos(\Omega_{\mathrm{orb}} t). \qquad (29)$$

These vectors were obtained by taking the derivative of the orbital time delay with respect to $\Omega_{\mathrm{orb}}$. Incorporating these vectors significantly reduces the number of required orbital period trials. For example, for the aforementioned parameters, the number of orbital period trials decreases from approximately $10^8$ to approximately $\sqrt{\frac{a}{\sigma P}}\frac{T}{P_{\mathrm{orb}}} \approx 10^6$. We can add more derivatives to the lattice, reducing the required enumeration. Unlike adding the first derivatives, when we add the second derivatives, the sensitivity can be compromised because not every point in the lattice is physical, and this can artificially increase the look-elsewhere effect. This is because the coefficients of the second derivative vectors are fully determined, in a nonlinear way, from the coefficients of the first. For example, the ratio between the coefficients of $v_{1,\mathrm{orb}}$ and $v_{2,\mathrm{orb}}$ is the same as between the coefficients of $v_{3,\mathrm{orb}}$ and $v_{4,\mathrm{orb}}$, because $v_{4,\mathrm{orb}}$ is a second derivative vector, with respect to the phase and orbital frequency.

Note that the spin-down vectors (corresponding to ($f^{(1)}$, $f^{(2)}$,...)) are correct when using the source time, which is inaccessible to us (we know only the observed time). This introduces a coupling between the spin-down parameters and the orbital parameters. Fortunately, the two vectors correcting these coefficients are the same as those compensating for orbital period change and small eccentricity changes.

An in-depth discussion on partitioning the enumeration space (including a full Keplerian orbit) into a set of linear spaces will be described in future work. In this paper, we discuss the case of a pulsar in a circular orbit.

## 4. Information Limit—Performance under the Null Hypothesis $H_0$

Understanding the expected performance without a signal is important when using this lattice-based solution as a detection tool. Therefore, we compute the expected length of the shortest nontrivial vector in a lattice constructed with random TOAs. A useful tool to analyze our lattice setup is the Gaussian heuristic (GH), which states that the probability of finding lattice sites in some volume is proportional to the volume, and the expected length (per coordinate) of the shortest vector in a lattice is

$$\lambda_1 = \frac{\mathrm{vol}(\mathcal{L})^{1/n}}{\sqrt{2\pi e}}, \qquad (30)$$

where $n$ is the lattice dimension and $\mathrm{vol}(\mathcal{L})$ is the lattice volume, calculated as

$$\mathrm{vol}(\mathcal{L}) = \sqrt{\det LL^T}\,. \qquad (31)$$

But, because the original lattice's volume is wholly controlled by $d_\phi$, $d_f$,..., the arbitrarily small numbers intended to make the timing vectors quasi-continuous, the GH is unsuitable for analyzing it.

Note that, once we set the coefficients for the unit vectors (or the number of integer revolutions for each TOA), we only need to perform a simple linear fit with respect to the timing vectors to find the smallest phase residuals. However, we can reverse the order, orthogonalize the unit vectors with respect to the timing vectors, and then perform the integer search. We will refer to this reverse order search as searching in the sublattice orthogonal to the timing vectors, and it is suitable for analysis using the GH.[12] We use the following lattice:

$$L = \begin{pmatrix} I_{n\times n} & \mathbb{O}_{n\times m} \\ V_{m\times n} & \eta_{m\times m} \end{pmatrix}, \qquad (32)$$

where

$$V^T_{m\times n} = \begin{pmatrix} d_1 \boldsymbol{v}_1 & \cdots & d_m \boldsymbol{v}_m \end{pmatrix},$$

and

$$(\eta_{m\times m})_{i,j} = \delta_{i,j} d_i \sigma_{\exp}/\sigma_i,$$

where $\boldsymbol{v}_i$ are the orthonormalized timing vectors and $d_i$ are small factors used to make the timing model vectors $\vec{v}_i$ arbitrarily small. $\sigma_{\exp}$ is the expected length of a random vector (which we will now solve for self-consistently), and $\sigma_i$ is the range that we search for in $v_i$ (similarly to $f_{\max}$ for the periodicity vector). Following these definitions, we can calculate the lattice volume (see Appendix for a detailed calculation):

$$\mathrm{vol}(\mathcal{L}) = \prod_i \left( \frac{\sigma_{\exp}/\sigma_i}{\sqrt{1 + (\sigma_{\exp}/\sigma_i)^2}} \right). \qquad (33)$$

Next, after substituting Equation (33) into Equation (30), we obtain

$$\sigma_{\exp} \equiv \lambda_1 = \left( \prod_i \frac{\sigma_{\exp}/\sigma_i}{\sqrt{1 + (\sigma_{\exp}/\sigma_i)^2}} \right)^{1/n} \Big/ \sqrt{2\pi e}\,. \qquad (34)$$

Solving for $\sigma_{\exp}$ generally involves solving a high-degree polynomial and requires a numerical treatment. But, in the typical case, we search for solutions with large ranges for the timing vectors (relative to a single phase cycle), allowing us to simplify the calculation:

$$\sigma_{\exp} \approx \sigma_{\exp}^{m/n} \left( \prod_i \sigma_i \right)^{-1/n} \Big/ \sqrt{2\pi e}\,, \qquad (35)$$

After solving Equation (35) for $\sigma_{\exp}$, we obtain:

$$\sigma_{\exp} \approx \left( \prod_i \sigma_i \right)^{-\frac{1}{n-m}} \Big/ \sqrt{2\pi e}^{\frac{n}{n-m}}. \qquad (36)$$

Knowing how to compute the minimum length of a spurious signal precisely, we can estimate the false-alarm probability (FAP) of a candidate signal with $\sigma_{\mathrm{cand}}$ by computing the expected number of lattice sites with $\sigma \leqslant \sigma_{\mathrm{cand}}$ using the GH. The estimated FAP is given by

$$\mathrm{FAP} = (\sigma_{\mathrm{cand}}/\sigma_{\exp})^n\,. \qquad (37)$$

## 5. Complexity Analysis

Following the same logic, we can compute the complexity (and the amount of data) needed to solve a timing problem with

[12] In solutions spanned by the orthogonal sublattice, some of the phase residuals might be larger than half a revolution, $|\varepsilon_i| > 0.5$.

**Table 1**
Complexity of the Lattice Sieve for Different Pulse Widths and Corresponding Association Probabilities

| $\sigma$ | $p(\text{at } \sigma_{\text{int}} = 0)$ | $C(\Lambda)$ | $C(\Lambda = 10^{28})$ | $n(\Lambda = 10^{28})$ |
|---|---|---|---|---|
| 0.2 | 0.5 | $\Lambda^{0.81}$ | $10^{22.5}$ | 207 |
| 0.16 | 0.7 | $\Lambda^{0.44}$ | $10^{12}$ | 111 |
| 0.11 | 0.85 | $\Lambda^{0.27}$ | $10^{7.5}$ | 69 |
| 0.06 | 0.95 | $\Lambda^{0.17}$ | $10^{4.7}$ | 42 |

**Note.** $\sigma$ is the pulse width, $p$ is the association probability of the TOAs, and $\sigma_{\text{int}}$ is the intrinsic pulse width; they are related by $\sigma^2 = p \cdot \sigma_{\text{int}}^2 + (1-p) \cdot \frac{1}{12}$.

a given entropy (number of "independent" options):

$$\Lambda \equiv \frac{1}{\text{vol}(\mathcal{L})} = \prod_i \frac{\sigma_i}{\sigma_{\text{exp}}} \sqrt{1 + (\sigma_{\text{exp}}/\sigma_i)^2}\,. \tag{38}$$

In the case where the correct solution is not the shortest vector,

$$\sigma_{\text{exp}} < \sigma, \tag{39}$$

we might still be able to find it by generating $N_{\text{candidates}}$ short vectors, while keeping the condition

$$\left(\frac{\sigma}{\sigma_{\text{exp}}}\right)^n \leqslant N_{\text{candidates}}. \tag{40}$$

For the lattice sieve, we use $N_{\text{candidates}} \approx 2^{0.2n}$ and the time complexity[13] is $C(n) \approx 2^{0.36n}$, as measured by L. Ducas et al. (2021) for $n \sim 100$.

In Table 1, we provide the time complexity of our method for different pulse widths and their corresponding association probabilities. We find that $\sigma < 0.11$ is easily attainable, while $p = 0.5$ is still unfeasible in this framework, even with an intrinsically infinitely narrow pulse.

## 6. Injection Recovery—Performance under the Alternative Hypothesis $H_1$

As in any detection problem, we need to analyze our algorithm in the presence of a signal, and in this case, we are presented with a challenge. Since we use the G6K lattice solver, which is designed to solve the SVP and not the shortest nontrivial vector problem by enumeration, the solver may not always find the shortest nontrivial vector. Therefore, we performed a basic injection-recovery analysis of an isolated MSP with a characteristic age of 100 Myr and a position known to a precision of 1° ($\Lambda_d \sim 10^{28}$).

We performed this analysis in both an FRB-like and a Fermi-LAT–like scenario. In the FRB-like scenario, we sieve in the sublattice that contains all noncontinuous vectors (not $d_f \boldsymbol{t}$, $d\boldsymbol{\phi}$, etc...) and compare its shortest nontrivial vector against the injected one. The results, shown in Figure 1, demonstrate that the $H_0$ limit can be reached but not surpassed.

In the Fermi-LAT–like scenario, we use only photons with high association probabilities in the lattice and later verify the solutions using photons with lower association probabilities. Therefore, we sieve to generate many candidate solutions (in the same sublattice as in the FRB scenario). If the injected signal is one of the candidates, we regard it as a successful recovery. The results from this analysis (see Figure 1) show that it is possible to significantly surpass the $H_0$ limit because we allow multiple trials.

## 7. Results on Real Data

Here, we present the application of this method to 4FGL J0318.2+0254, a Fermi-LAT source also known as PSR J0318+0253, discovered using the FAST radio telescope (P. Wang et al. 2021). 4FGL J0318.2+0254 is an isolated MSP with a sufficient number of high-probability photons and a narrow enough pulse. We used the Fermi-LAT data from 2008 July 31 to 2023 September 28 and selected photons arriving within 3° of the source's position in 4FGL-DR4 (J. Ballet et al. 2023). We assigned association probabilities to the photons using the standard fermitools procedure (Fermi Science Support Development Team 2019). We divided the photons into two sets based on their association probabilities: the 70 highest-probability photons and the rest with a lower limit of $p \geqslant 0.2$, which we now refer to as the lattice set and verify set, respectively. We construct the lattice using the lattice set of high-probability photons, searching for $f \sim 100$ Hz and $\tau \sim 100$ Myr, assuming the position reported in 4FGL-DR4 (with errors equal to the 95% confidence interval uncertainties) and a proper motion on the order of 10 mas yr$^{-1}$. We reduced the lattice and generated $\sim 6.5 \times 10^4$ short candidates. From these candidates, we selected the ones with high enough $f$ and used them to fold the verify set, calculating the H-test statistic (P. Bickel et al. 2008) for each candidate. The results from this analysis are shown in Figure 2. Using the H-test scores of the verify set of photons, we identified the correct solutions. The phase-folded histogram from one of these correct solutions is shown in Figure 3.[14]

## 8. The Norm Problem—Adapting to Bad Photons and Double Pulse Profiles

The most severe conceptual problem in our setup is the fact that we are using the $L_2$ norm to decide between the different possible solutions.

The $L_2$ norm corresponds to the assumption of a perfect Gaussian pulse profile, which does not hold for a vast majority of known $\gamma$-ray pulsars that tend to have a double pulse profile. This is also a bad norm to use when background photons are present, which is the typical case for sources in Fermi-LAT.

A better choice of a test statistic to rank the different lattice vectors is the H-test (P. Bickel et al. 2008). The H-test corrects for both a somewhat more general pulse shape and for the fact that different TOAs have different association probabilities with the source (at least in the Fermi-LAT case).

Moreover, it is useful to output many vectors from the lattice sieve and rank them according to the more sensitive H-test, thereby picking up the correct solution even if the $L_2$ norm ranks the correct solution only in the $M^{\text{th}}$ place. Sieving algorithms are usually using a large number of vectors, and this approach is beneficial to increase the sensitivity of the search.

Additionally, the requirement for a large number ($N > 60$) of photons with high association probability ($p > 0.85$) limits the applicability of the method to a few dozen sources (out of the thousands of unassociated sources). In a follow-up paper, we

[13] Time complexity refers to the number of basic computer operations required.

[14] A notebook that roughly follows this procedure is available at https://github.com/Zackay-Lab/pulsar-lattice-example.

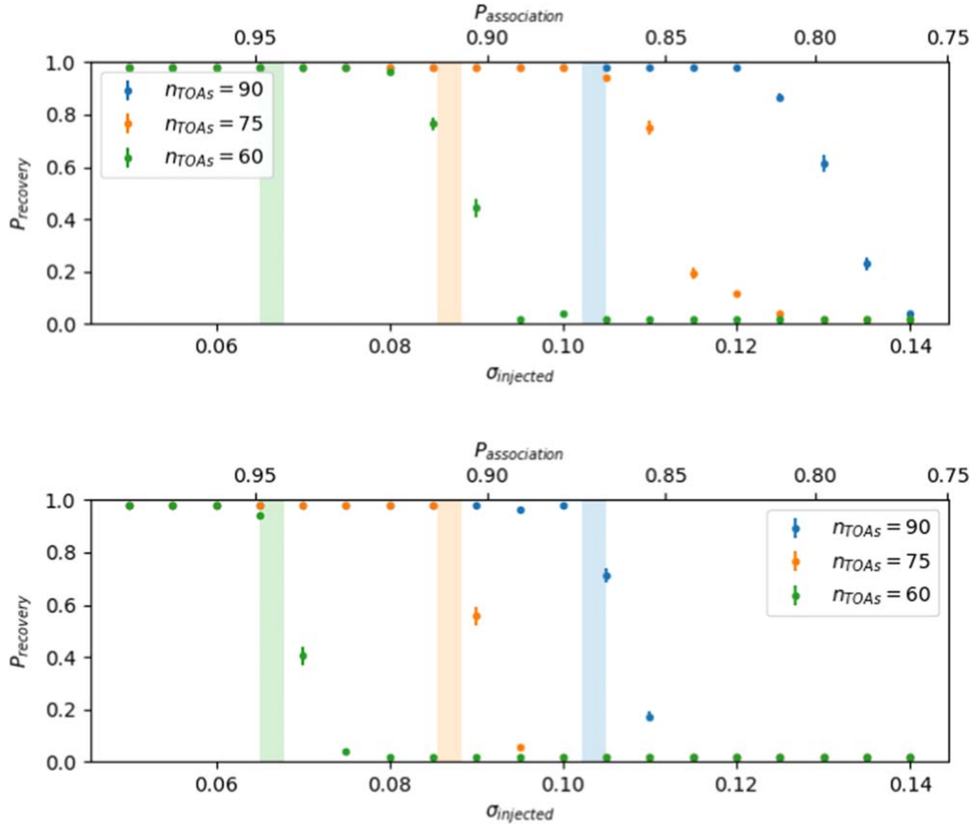

**Figure 1.** The probability for recovery as a function of the injected signal's width $\sigma$. Top: Fermi-LAT–like simulation with an additional verification step. Bottom: FRB-like simulation with no additional verification step. Recovery probability was calculated based on 50 injections per $\sigma$ for different numbers of TOAs over 10 yr. The simulation consisted of an isolated MSP ($f \in [100, 1000]$ Hz and $\tau = 100$ Myr). The expected information limit, $\sigma_{\rm exp}$ from Section 4, is indicated using the shaded vertical stripes. The information limits appear as a stripe, rather than a single line, because the entropy depends on the specific simulated sky position. A sharp transition from constant to no recovery at and after $\sigma_{\rm exp}$ is seen in both the FRB-like and Fermi-LAT–like simulations. An important distinction between the two simulations is that we check whether the solution is in the short vectors set in the Fermi-LAT-like simulation, while we check whether the solution is the *shortest* in the short vectors set in the FRB-like simulation.

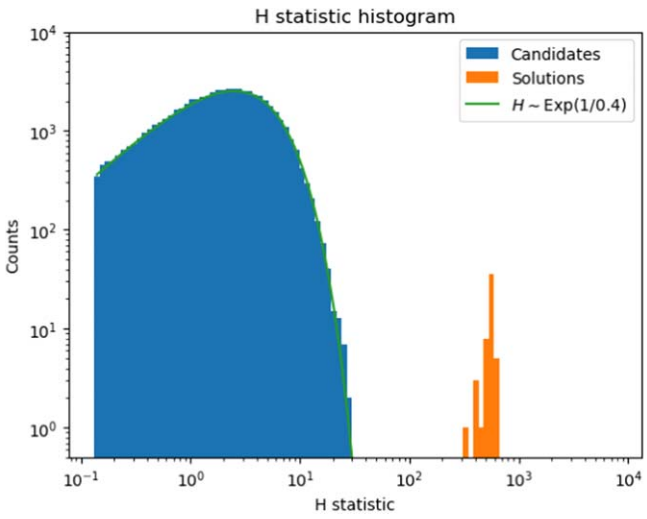

**Figure 2.** The H-test statistic histogram for significant and insignificant candidates (which we will call solutions and candidates, respectively) generated using the lattice sieving. The H-test is calculated over the verify set of photons, and the significance threshold for the $p$-value is set at $10^{-7}$. The green line corresponds to the distribution of an exponential random variable with parameter $\lambda = 1/0.4$, as expected from the simulation by O. De Jager & I. Büsching (2010).

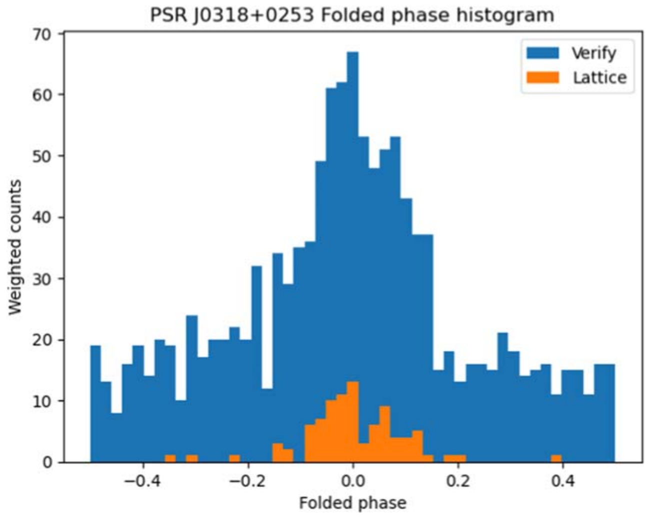

**Figure 3.** Weighted histograms of one of the solution's phase fold for the lattice-set and verify-set photons.

will present an algorithmic improvement that will more effectively recover the correct solution in the situation of only moderate association probabilities ($p \sim 0.5$).

## 9. Conclusions

In this paper, we have shown that the timing solutions of pulsars can be recovered using lattice algorithms, a set of advanced tools developed for cryptanalysis. We have also demonstrated that a lattice-based approach can be used to solve a timing model that is a linear combination of known vectors. Additionally, we showed that lattice algorithms can solve problems that were previously impossible to solve, in a matter of seconds.

We discussed the computational complexity of the lattice algorithm technique and showed that it is substantially smaller than full enumeration of the parameter space. The results strongly depend on the duty cycle (or the equivalent width, for more complex pulse shapes) and the association probabilities of the arrival times.

As a proof of concept, we recovered the timing solution of a real pulsar using Fermi-LAT data and our lattice algorithm implementation. In the next papers in this series, we will show how to linearize a Keplerian orbit and present a novel algorithm that is more computationally efficient, allowing us to solve lattice problems in which half of the TOAs are random or cases where the pulsar has a double pulse profile. We will then apply our methods to all relevant Fermi-LAT unassociated sources.

## Acknowledgments

A.B.P. is a Banting Fellow, a McGill Space Institute (MSI) Fellow, and a Fonds de Recherche du Quebec—Nature et Technologies (FRQNT) postdoctoral fellow. B.Z. is supported by a research grant from the Willner Family Leadership Institute for the Weizmann Institute of Science.

This research was supported by the Minerva Foundation with funding from the Federal German Ministry for Education and Research. This research is supported by the Israeli Council for Higher Education (CHE) via the Weizmann Data Science Research Center, and by a research grant from the Estate of Harry Schutzman.

We greatly thank Tejaswi Venumadhav, Liang Dai, Matias Zaldarriaga, Martin Albrecht, Leo Ducas, Adi Shamir, Zvika Brakerski, Ittai Rubinstein, Asaf Shuv, Nathan Keller, Noga Bashan, and Doron Walach for in-depth discussions regarding various aspects of this project.

## Appendix
## Lattice Volume Calculation

According to the Gaussian heuristic (GH), the expected length of the shortest vector in a lattice is (per coordinate)

$$\lambda_1 = \frac{\mathrm{vol}(\mathcal{L})^{1/n}}{\sqrt{2\pi e}}, \tag{A1}$$

where $n$ is the lattice dimension and $\mathrm{vol}(\mathcal{L})$ is the lattice volume, calculated as

$$\mathrm{vol}(\mathcal{L}) = \sqrt{\det LL^T}. \tag{A2}$$

In our setup, we are interested in the sublattice consisting of the unit vectors of the different TOAs, projected orthogonally to the quasi-continuous timing model vectors, the phase residuals lattice. We use the following lattice:

$$L = \begin{pmatrix} I_{n\times n} & \mathbb{O}_{n\times m} \\ V_{m\times n} & \eta_{m\times m} \end{pmatrix}, \tag{A3}$$

where

$$V^T_{m\times n} = \begin{pmatrix} d_1\boldsymbol{v}_1 & \cdots & d_m\boldsymbol{v}_m \end{pmatrix};\ \boldsymbol{v}_i \cdot \boldsymbol{v}_j = \delta_{ij},$$

and

$$(\eta_{m\times m})_{i,j} = \delta_{i,j} d_i \sigma_{\exp}/\sigma_i,$$

where $\vec{v}_i$ are the orthonormal timing vectors, $d_i$ are small constants, $\sigma_i$ are the extent of the timing vectors we wish to search over, and $\sigma_{\exp}$ is the length per coordinate of the shortest random vector we are trying to estimate. We wish to project out the $\begin{pmatrix} V_i & \eta_i \end{pmatrix}$ directions, and for that, it would be useful to define the unscaled versions of $V$ and $\eta$:

$$U^T \equiv \begin{pmatrix} \vec{v}_1 & \cdots & \vec{v}_m \end{pmatrix};\ \Sigma \equiv \delta_{i,j}\sigma_{\exp}/\sigma_i. \tag{A4}$$

This can be done by using the projection operator:

$$P_V = I_{n+m\times n+m} - \begin{pmatrix} U & \Sigma \end{pmatrix}^T \left(\begin{pmatrix} U & \Sigma \end{pmatrix}\begin{pmatrix} U & \Sigma \end{pmatrix}^T\right)^{-1}\begin{pmatrix} U & \Sigma \end{pmatrix}. \tag{A5}$$

The projected sublattice can then be written as

$$\Pi_{n\times n+m} = I_{n\times n+m} P_V, \tag{A6}$$

and the matrix we are interested in its determinant is

$$\begin{aligned} \Pi\Pi^T &= I_{n\times n+m} P_V P_V I_{n+m\times n} \\ &= I_{n\times n+m} P_V I_{n+m\times n} \\ &= I_{n\times n} - U^T (I_{m\times m} + \Sigma\Sigma^T)^{-1} U. \end{aligned} \tag{A7}$$

Now, to calculate the determinant, we use the Weinstein–Aronszajn identity:

$$\det \Pi\Pi^T = \det I_{n\times n} - U^T (I_{m\times m} + \Sigma\Sigma^T)^{-1} U \tag{A8}$$

$$= \det I_{m\times m} - (I_{m\times m} + \Sigma\Sigma^T)^{-1} UU^T \tag{A9}$$

$$= \det I_{m\times m} - (I_{m\times m} + \Sigma\Sigma^T)^{-1}. \tag{A10}$$

If $\Sigma$ is diagonal, this reduces to

$$\det \Pi\Pi^T = \prod_i \frac{\Sigma_i^2}{1 + \Sigma_i^2}, \tag{A11}$$

and the volume is

$$\mathrm{vol}(\mathcal{L}) = \prod_i \frac{\Sigma_i}{\sqrt{1 + \Sigma_i^2}}. \tag{A12}$$

## ORCID iDs

Dotan Gazith https://orcid.org/0000-0001-6698-3693
Aaron B. Pearlman https://orcid.org/0000-0002-8912-0732
Barak Zackay https://orcid.org/0000-0001-5162-9501

## References

Abdollahi, S., Acero, F., Baldini, L., et al. 2022, ApJS, 260, 53
Albrecht, M. R., Ducas, L., & Herold, G. 2019, in Advances in Cryptology – EUROCRYPT 2019, ed. Y. Ishai & V. Rijmen (Cham: Springer), 717
Allen, B., Knispel, B., Cordes, J. M., et al. 2013, ApJ, 773, 91
Antoniadis, J., Arzoumanian, Z., Babak, S., et al. 2022, MNRAS, 510, 4873
Atwood, W. B., Ziegler, M., Johnson, R. P., & Baughman, B. M. 2006, ApJ, 652, L49
Babai, L. 1986, Combinatorica, 6, 1
Bailes, M. 2022, Sci, 378, abj3043
Ballet, J., Bruel, P., Burnett, T., & Lott, B. 2023, arXiv:2307.12546

Becker, A., Gama, N., & Joux, A. 2015, Cryptology ePrint Archive, Paper 2015/522, https://eprint.iacr.org/2015/522
Bhardwaj, M., Michilli, D., Kirichenko, A. Y., et al. 2024, ApJL, 971, L51
Bickel, P., Kleijn, B., & Rice, J. 2008, ApJ, 685, 384
Bruzewski, S., Schinzel, F. K., & Taylor, G. B. 2023, ApJ, 943, 51
Clark, C. J., Breton, R. P., Barr, E. D., et al. 2023, MNRAS, 519, 5590
Cordes, J. M., & Chatterjee, S. 2019a, ARA&A, 57, 417
Coronado-Blázquez, J. 2022, MNRAS, 515, 1807
De Jager, O., & Büsching, I. 2010, A&A, 517, L9
Du, C., Huang, Y.-F., Zhang, Z.-B., et al. 2024, ApJ, 977, 129
Ducas, L. 2018, in Advances in Cryptology – EUROCRYPT 2018, ed. J. B. Nielsen & V. Rijmen (Cham: Springer), 125
Ducas, L., Stevens, M., & van Woerden, W. 2021, in Advances in Cryptology – EUROCRYPT 2021, ed. A. Canteaut & F. X. Standaert (Cham: Springer), 249
Fermi Science Support Development Team 2019, Fermitools: Fermi Science Tools, Astrophysics Source Code Library, ascl:1905.011
Fonseca, E., Andersen, B. C., Bhardwaj, M., et al. 2020, ApJL, 891, L6
Frail, D. A., Ray, P. S., Mooley, K. P., et al. 2018, MNRAS, 475, 942
Gama, N., Nguyen, P. Q., & Regev, O. 2010, in Advances in Cryptology – EUROCRYPT 2010, ed. H. Gilbert (Cham: Springer), 257
Germani, S., Tosti, G., Lubrano, P., et al. 2021, MNRAS, 505, 5853
Khot, S. 2004, in 45th Annual IEEE Symp. on Foundations of Computer Science, ed. D. Azada (Piscataway, NJ: IEEE), 126
Kirsten, F., Marcote, B., Nimmo, K., et al. 2022, Natur, 602, 585
Lenstra, A. K., Lenstra, H. W., & Lovász, L. 1982, MatAn, 261, 515
Li, D., Wang, P., Zhu, W. W., et al. 2021, Natur, 598, 267
Lorimer, D. R. 2008, LRR, 11, 1
Lorimer, D. R., & Kramer, M. 2004, in Handbook of Pulsar Astronomy, ed. R. Ellis et al., Vol. 4 (Cambridge: Cambridge Univ. Press)
Luo, J., Ransom, S., Demorest, P., et al. 2021, ApJ, 911, 45
Macquart, J. P., Prochaska, J. X., McQuinn, M., et al. 2020, Natur, 581, 391
Majid, W. A., Pearlman, A. B., Prince, T. A., et al. 2021, ApJL, 919, L6
Malyshev, D. V., & Bhat, A. 2023, MNRAS, 521, 6195
McLaughlin, M. A., Lyne, A. G., Keane, E. F., et al. 2009, MNRAS, 400, 1431
Mickaliger, M. B., McLaughlin, M. A., Lorimer, D. R., et al. 2012, ApJ, 760, 64
Neyman, J., & Pearson, E. S. 1933, RSPTA, 231, 289
Nguyen, P. Q., & Vidick, T. 2008, J. Math. Crypt., 2, 181
Nieder, L., Allen, B., Clark, C. J., & Pletsch, H. J. 2020a, ApJ, 901, 156
Nieder, L., Clark, C. J., Kandel, D., et al. 2020b, ApJL, 902, L46
Nimmo, K., Hessels, J. W. T., Kirsten, F., et al. 2022, NatAs, 6, 393
Niu, J.-R., Zhu, W.-W., Zhang, B., et al. 2022, RAA, 22, 124004
Pearlman, A. B., Majid, W. A., Prince, T. A., Kocz, J., & Horiuchi, S. 2018, ApJ, 866, 160
Pearlman, A. B., Scholz, P., Bethapudi, S., et al. 2024, NatAs
Petroff, E., Hessels, J. W. T., & Lorimer, D. R. 2019, A&ARv, 27, 4
Phillips, C., & Ransom, S. 2022, AJ, 163, 84
Platts, E., Weltman, A., Walters, A., et al. 2019, PhR, 821, 1
Pletsch, H. J., & Clark, C. J. 2014, ApJ, 795, 75
Prager, B. J., Ransom, S. M., Freire, P. C. C., et al. 2017, ApJ, 845, 148
Schnorr, C., & Euchner, M. 1994, MatPr, 66, 181
Snelders, M. P., Nimmo, K., Hessels, J. W. T., et al. 2023, NatAs, 7, 1486
Stairs, I. H. 2003, LRR, 6, 5
Teunissen, P. J. 1993, in IAG general meeting: Invited lecture, section IV theory and methodology
The CHIME/FRB Collaboration, Amiri, M., Andersen, B. C., et al. 2020, Natur, 582, 351
The CHIME/FRB Collaboration, Andersen, B. C., Bandura, K., et al. 2019, ApJL, 885, L24
The CHIME/FRB Collaboration, Andersen, B. C., Bandura, K., et al. 2022, Natur, 607, 256
The CHIME/FRB Collaboration, Andersen, B. C., Bandura, K., et al. 2023, ApJ, 947, 83
The FPLLL development team 2024, fpylll, a Python wrapper for the fplll lattice reduction library, v0.6.2, https://github.com/fplll/fpylll
Walters, A., Weltman, A., Gaensler, B. M., Ma, Y.-Z., & Witzemann, A. 2018, ApJ, 856, 65
Wang, P., Li, D., Clark, C. J., et al. 2021, SCPMA, 64, 129562
Xu, H., Niu, J. R., Chen, P., et al. 2022, Natur, 609, 685
Zhu, K. R., Chen, J. M., Zheng, Y. G., & Zhang, L. 2024, MNRAS, 527, 1794