Dynamical Decoupling using Universal Optimal Tracking

Amit Devra,[1,2,*] Emanuel Malvetti,[1,2,†] Niklas J. Glaser,[1,3] Abhishek Agarwal,[4] Ivan Rungger,[4,5] Santana Lujan,[6] Max Werninghaus,[3] Stefan Filipp,[1,3] Léo Van Damme,[1,2,‡] and Steffen J. Glaser[1,2,§]

[1] *Technical University of Munich, TUM School of Natural Sciences, Garching 85748, Germany*
[2] *Munich Center for Quantum Science and Technology (MCQST), Schellingstraße 4, München 80799, Germany*
[3] *Walther-Meißner-Institut, Bayerische Akademie der Wissenschaften, Garching 85748, Germany*
[4] *National Physical Laboratory, Teddington, TW11 0LW, United Kingdom*
[5] *Department of Computer Science, Royal Holloway, University of London, Egham, TW20 0EX, UK*
[6] *German Aerospace Center (DLR), Weßling, Germany*

(Dated: November 10, 2026)

Dynamical decoupling (DD) is a widely used and resource-efficient technique for error suppression, but conventional DD relies on periodically repeating a short pulse block to refocus the qubit state during idle periods. Imperfections in this block cause residual errors to accumulate, ultimately degrading state recovery over long idle times. Here, we introduce a universal optimal tracking approach that extends the original tracking concept to a fully state-independent setting for designing DD sequences. By monitoring the qubit's evolution at predefined waypoints during optimization, the method dynamically compensates residual errors while preserving regular refocusing. Experimental demonstrations on a superconducting-qubit platform confirm the suppression of error accumulation under static control imperfections, in agreement with numerical predictions. Complementary simulations further show that optimal-tracking-based sequences maintain strong performance under time-dependent noise. These results establish optimal tracking as a practical and hardware-agnostic approach to designing short, robust DD sequences suitable for noisy quantum devices.

arXiv:2606.21762v3 [quant-ph] 9 Nov 2026

I. INTRODUCTION

Quantum systems are inherently fragile and highly susceptible to environmental noise, leading to decoherence and posing a significant challenge for their reliable operation [1–4]. Dynamical decoupling (DD) is among the earliest and most widely adopted techniques for suppressing errors in quantum systems. Building on the concept of spin echoes developed by Erwin Hahn in 1950 [5], the elimination of unwanted interactions has been the subject of various decoupling techniques [6–10]. DD has been successfully implemented across a variety of quantum platforms, including superconducting qubits [4, 11–14], trapped ions [15, 16], neutral atoms [17], photonic qubits [18], and quantum sensing [19, 20]. The core principle of DD is to preserve quantum coherence by applying precisely timed control pulses that average out unwanted system-environment interactions [7, 21, 22]. In today's noisy quantum computers, it has become a powerful tool for enhancing the fidelity of quantum algorithms [4, 23–27], and is now routinely incorporated into circuit compilation routines for various hardware platforms [28].

Over the years, numerous DD sequences have been developed, each offering different levels of protection against specific types of noise, as reviewed in the studies [12, 29, 30]. For instance, the CPMG sequence is known not to be universal [10, 12], meaning it does not protect all quantum states equally. In contrast, sequences such as XY4 are universal and also robust up to some order against pulse imperfection errors such as drive frequency deviations (detuning) and pulse-amplitude deviations (over- or under-rotations) [12]. Basic DD sequences simply repeat the same pulse or block of pulses throughout the decoupling sequence. However, in practical scenarios where the pulse block does not perfectly refocus the qubit state (e.g., due to imperfect pulses, the net propagator deviates away from identity), residual errors can accumulate across successive repetitions, leading to poor state recovery at the end of the idle time. Consequently, enhancing the robustness of these sequences remains a key objective in the design of dynamical decoupling protocols. [22].

To address this challenge, several strategies have been proposed, such as using composite pulses [31–33], which give rise to sequences like KDD [10], or by concatenating existing sequences, as in CDD [34, 35]. However, these approaches often lead to an exponential increase in the number of required pulses to achieve certain robustness [12, 29]. The universal robust (UR) family of DD sequences addresses this challenge by providing robustness to a broader range of pulse imperfections with only N pulses per block [22]. Notably, low-order UR sequences reduce to well-established DD sequences (e.g., for $N = 2$, UR2 is identical to CPMG and for $N = 4$, UR4 is identical to XY4), indicating that their robustness is similarly limited for small N. We note that the term 'universal robust' (UR) refers to robustness against a broad class of pulse imperfections, and should not be confused

* amit.devra@tum.de; Present address: Munich Quantum Valley gGmbH, Walther-von-Dyck-Straße 6, 85748 Garching, Germany
† Present address: Zurich Instruments AG, Technoparkstrasse 1, 8005 Zurich, Switzerland
‡ leo.van-damme@tum.de
§ glaser@tum.de

with 'universal rotation' pulses [32, 36, 37], which denote pulses that produce the same rotation angle regardless of resonance offset or rf-amplitude inhomogeneity. Throughout this paper, UR refers exclusively to universal robust DD sequences. A complementary approach exploits supercycle constructions, where short four-pulse blocks such as XY4 or MLEV-type sequences are combined with cyclic phase shifts and sign reversals (e.g., XY64 and related schemes) to enhance robustness while retaining a four-pulse structure [8, 9, 38]. A related strategy is to randomize the order of the control operations from block to block, which slows the accumulation of residual averaging errors in long sequences [39, 40]. In practice, however, shorter base sequences are often essential in large-scale quantum circuits, where idle windows of varying duration arise between gate operations. Such short DD blocks can be inserted into these temporal gaps without disrupting the surrounding control operations. Moreover, short sequences enable more frequent refocusing, i.e., periodically returning the qubit state close to its initial state, which improves resilience to time-dependent noise, a major source of decoherence arising from fluctuating system-environment coupling, for example, due to two-level fluctuators [41, 42]. These considerations highlight the need for DD sequences that are both short and intrinsically robust, not only against static detuning errors and pulse imperfections but also against time-dependent noise, so that they continue to average out unwanted dynamics even when the environment is not stationary.

In this work, we introduce a new family of dynamical decoupling sequences, denoted as U-TRACK, based on the idea of *Universal optimal TRACKing*. The key idea of tracking is to enforce, during optimization, that the qubit's evolution approximates the identity not only at the end of the sequence but also at intermediate checkpoints, i.e., after each pulse block, thereby preventing error accumulation across successive cycles. The optimal tracking framework was originally developed to efficiently decouple two heteronuclear spins in NMR [43, 44], where a GRAPE-based algorithm [45] was used to optimize (non-periodic) decoupling sequences that transfer a defined initial state to a desired target state not only at the end of the sequence (the final time) but also at specified tracking points during the sequence. Here, we generalize this concept to a universal control setting, i.e., one that is independent of the input state, to design dynamical decoupling sequences. This approach enables a design principle that fundamentally differs from conventional DD approaches, as illustrated in Fig. 1.

This approach enables the design of a DD sequence that provides dynamic compensation to suppress error accumulation. Unlike basic DD sequences, which rely on repeating identical pulse blocks and thus allow systematic errors to accumulate, the resulting U-TRACK

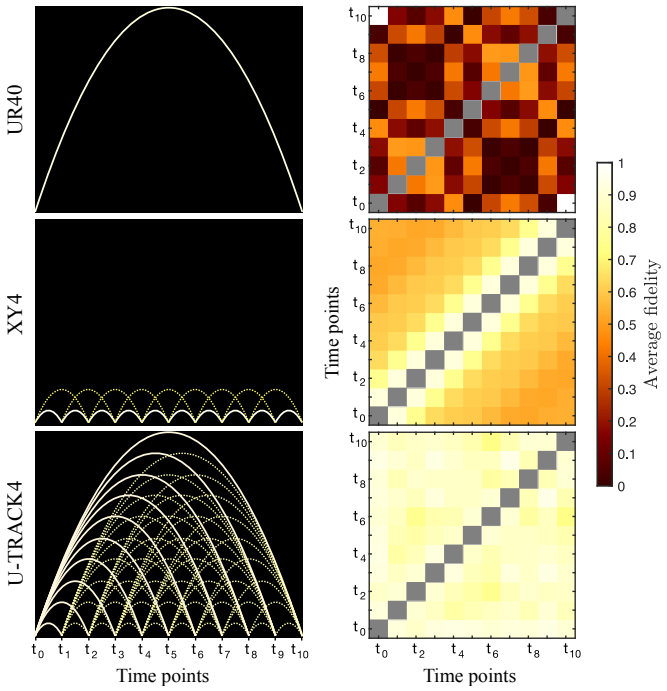


FIG. 1. Comparison of the design principles of two conventional DD sequences (UR40, top row, and XY4, middle row) with the novel tracking-based approach introduced in this paper (U-TRACK4, bottom row). On the left, the white arcs represent propagators $U(t_m, t_n)$ designed to approximate the identity operator $\mathbb{I}$ from time point t_m to time point t_n for $0 \leq m < n \leq 10$. Yellow dashed lines indicate propagators that, by direct transitivity, also approximate the identity operator. On the right, fidelities measuring the proximity of $U(t_m, t_n)$ to $\mathbb{I}$, averaged over a range of frequency detunings and pulse-amplitude deviations, are shown. Trivial diagonal entries with $U(t_n, t_n) = \mathbb{I}$ are grayed out. The vertical axis of the left panels is omitted as it carries no physical meaning; only the extent of the arcs is relevant.

sequences introduce corrections at each block, yielding non-periodic pulse structures. Consequently, robustness against parameter deviations, such as detuning and amplitude errors, remains stable even as the number of pulse blocks increases. This effect can be directly observed in the fidelity plots on the right of Fig. 1, where each entry (t_m, t_n) shows the average process fidelity of the propagator $U(t_m, t_n)$ with respect to the identity, averaged over a range of detuning and amplitude errors. U-TRACK4 maintains high fidelity at all intermediate times, reflected by bright entries throughout the matrix, while XY4 shows gradual degradation visible as darkening entries away from the diagonal. For longer sequences such as UR40 [22], where the entire sequence is optimized as a single identity gate between the initial and final times, fidelity is high at the endpoint by design, but intermediate-time fidelity is unconstrained and typically poor, as visible in the irregular fidelity matrix of UR40 in Fig. 1. As a result, any sudden change in the dynamics not covered by the model, such

as a mid-sequence parameter jump, will degrade fidelity even at the endpoint, as explicitly discussed later. We note that this block-by-block tracking structure is closely related to supercycle-based DD (e.g., XY64), but U-TRACK generalizes these constructions by optimizing arbitrary blocks and robustness regions. Moreover, the optimized U-TRACK sequences do not simply recover known supercycles: the optimization naturally converges to a structure — which we term XUR — that mixes MLEV-type and XY-type blocks within a single sequence, a combination not previously considered in the literature [8, 9, 46].

Our optimal tracking approach optimizes the pulse phases of each block using a combination of GRAPE-based gradient optimization and a genetic algorithm. The GRAPE component efficiently computes local gradients, while the genetic algorithm performs a structured global search across the pulse space, helping to avoid suboptimal local minima and increasing the likelihood of finding globally optimal solutions. The method supports an arbitrary number of blocks, M, and any even number of pulses, N, per block. To highlight the novelty and practical advantages of the tracking approach, we focus on the $N = 4$ sequence U-TRACK4 and investigate its robustness to parameter deviations both numerically and experimentally on superconducting-qubit hardware, comparing its performance against XY4 and UR40. The experimentally obtained robustness profiles show excellent agreement with the numerical results, as detailed in Sec. II B.

In Sec. II C, we examine the robustness of different DD protocols to time-dependent detuning noise and evaluate their average fidelity across an ensemble of random-telegraph-noise realizations, enabling a direct comparison of sequence performance as a function of switching rate. In Sec. III, we analyze the structure of the optimized sequences and show that they converge to a novel XUR form (a mixture of MLEV-type and XY-type blocks) and discuss how this structure can serve as an effective warm-start for the optimization, which in practice tends to reduce the optimization time. The optimization framework used in this work is available as a Julia package, UTrack.jl [47], and ready-to-use optimized sequences for various lengths N are provided in the package repository. The resulting sequences are hardware-agnostic; experimental validation in this study was performed on a superconducting quantum device.

II. OPTIMAL TRACKING

This section introduces the optimal tracking approach and compares it with existing DD methods. We first provide a qualitative overview of how the three approaches — single-block optimization, periodic repetition, and optimal tracking — differ in their design philosophies and practical performance. To make these differences concrete, we intentionally introduce experimental imperfections, such as static detuning and pulse-amplitude errors, and compare the resulting fidelities both numerically and experimentally. Full technical details on the U-TRACK optimization algorithm, the experimental implementation on a transmon-based quantum processor, and the transmon qubit simulations are provided in Appendices A, B, and C respectively.

A. Dynamical decoupling schemes

A dynamical decoupling (DD) sequence protects a qubit's state during idle periods — times when the qubit is not actively used — by applying a series of control pulses that counteract noise and environmental disturbances. Without DD, fluctuations in the qubit's environment can cause its state to drift, introducing errors into quantum-circuit outcomes. The two dominant error types are frequency detuning — which induces unwanted rotations about the z-axis — and pulse-amplitude deviations — which lead to over- or under-rotations around a transverse axis. On transmon-based devices, these errors may arise from imperfect calibration [48, 49], crosstalk [50], interactions with two-level systems (TLSs) [42, 51], flux noise [52] or pulse distortions [48, 53]. A DD sequence is designed so that, despite these imperfections, the qubit's net evolution during an idle time T is effectively the identity, i.e., $U(T) = \mathbb{I}$, where $U(T)$ denotes the qubit's evolution operator.

We compare three dynamical decoupling techniques:

(i) A single sequence of pulses spanning the entire idle period, such as UR40.

(ii) Repeated application of a shorter base sequence, such as XY4.

(iii) Optimal tracking (U-TRACK): a long sequence designed to periodically refocus the qubit state while suppressing error accumulation across blocks.

Here, U-TRACK optimizes a sequence of gates, namely π-rotations with optimized phases, to enable a direct comparison with existing DD methods. The method is, however, not limited to π-gates: general pulse-phase shaping could provide more degrees of freedom, which we leave for future work.

Approach (i) enables the design of sequences that are robust against large deviations in the Hamiltonian parameters, since the additional pulse energy available during longer idle periods allows for greater error compensation. In ref. [22], analytical formulas are provided to construct sequences that can span arbitrarily long idle times. These sequences, referred to as the universal robust (UR) family, are composed of multiple π-pulses with specific phases, and become increasingly robust against detuning and amplitude errors as the idle period

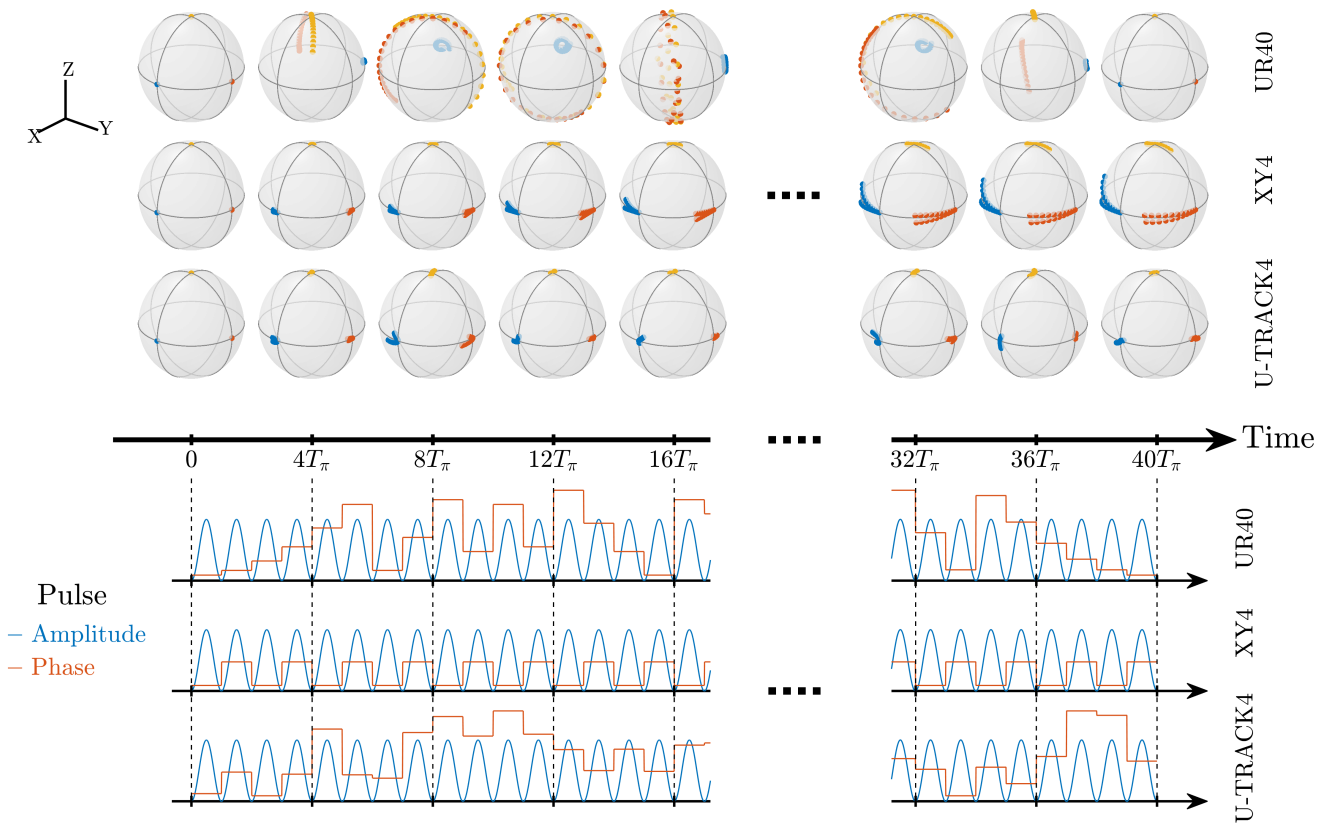


FIG. 2. Effect of three dynamical decoupling sequences on qubit dynamics, shown on the Bloch sphere (top) and as pulse waveforms (bottom). Each row corresponds to one sequence (UR40, XY4, U-TRACK4); each sphere is a snapshot in time, progressing from $t = 0$ to $t = 40\,T_\pi$, where T_π is the duration of one π-pulse. Points mark qubit states initialized on X (blue), Y (red), and Z (yellow). Each evolves under a different detuning strength; a tight cluster therefore indicates resilience to detuning errors, while a spread or displaced set of points indicates sensitivity. Ideally, every point returns to its starting position by $T = 40\,T_\pi$. UR40 achieves a very robust state recovery at T but not at intermediate times. XY4 recovers states after every block of four π-pulses, but small residual errors accumulate across repetitions, visible as a gradual drift of the point clusters. U-TRACK4, derived via optimal tracking control, matches XY4's block-by-block recovery while suppressing error accumulation, keeping the clusters tightly concentrated throughout the entire sequence.

— and correspondingly the number of π-pulses — grows, assuming these errors remain constant in time. However, while this assumption generally holds for a few gates acting on a qubit, additional events during longer idle periods may alter the Hamiltonian, reducing the sequence's effectiveness.

In contrast, the self-repeating structure of approach (ii) enables periodic refocusing of the qubit state. This repeated refocusing reduces sensitivity to time-dependent parameter fluctuations compared to approach (i). However, because it relies on repeating a short sequence with limited intrinsic robustness, residual errors accumulate across successive cycles, reducing robustness to parameter deviations during long idle periods.

Approach (iii) combines the strengths of approaches (i) and (ii) while overcoming their respective limitations. These sequences are referred to as the U-TRACK family. As in approach (i), a U-TRACK sequence is optimized over an arbitrarily long time, but it also refocuses the qubit state regularly, as in approach (ii). This optimization prevents error accumulation across cycles, yielding a sequence that remains effective in the presence of time-dependent fluctuations and robust against large parameter deviations, even over long idle periods.

One can also consider a fourth approach: repeating short DD sequences with small modifications between blocks, such as MLEV sequences or generalizations of XY4 [8, 9, 46, 54]. For example, XY64 combines several variations of XY4, where pulses are ordered as XYXY or (-X)(-Y)(-X)(-Y), and each can be cyclically shifted to YXYX or (-Y)(-X)(-Y)(-X). By construction, these sequences also regularly refocus the qubit state. These sequences share structural properties with optimal tracking, as discussed in Sec. III. Nevertheless, this

technique is limited to blocks of four π-pulses, whereas U-TRACK or UR pulses can support any even block length.

To illustrate the three approaches, we consider three DD schemes, namely UR40, XY4, and U-TRACK4, which correspond to approaches (i), (ii), and (iii), respectively. The sequences are illustrated in Fig. 2. UR40 consists of 40 π-pulses with phases computed using the analytical formula in Ref. [22]. XY4 is a sequence of four π-pulses with phases $\left(0, \frac{\pi}{2}, 0, \frac{\pi}{2}\right)$, repeated 10 times to cover the same idle period, and U-TRACK4 is a sequence of 40 π-pulses with phases optimized using optimal tracking, taken as the first 40 pulses of a sequence optimized over 100 blocks (Appendix A). Figure 2 illustrates the resulting qubit dynamics for states X, Y, and Z, under experimentally realistic detuning errors. We observe that UR40 recovers the state very accurately at the end of the full sequence, but some states are widely spread over the Bloch sphere at intermediate times. In contrast, XY4 recovers the states after each cycle of four pulses; however, because it is not perfectly recovered after a single cycle, the states slowly drift over successive cycles. The U-TRACK4 sequence combines the advantages of both techniques: it refocuses the states after every block of four π-pulses while avoiding error accumulation. We note that supercycle-based sequences such as XY64 exhibit behavior similar to U-TRACK4 for four-pulse blocks, as discussed in Appendix D.

B. Robustness to parameter deviations

The differences in robustness between the three dynamical decoupling approaches are illustrated in Fig. 3. In this figure, amplitude deviations ϵ and static detuning δ were intentionally applied on the quantum device to mimic experimental imperfections, and the fidelities of each DD sequence were computed from quantum process tomography measurements [55] for every (ϵ, δ) pair as described in Appendix B. Quantum process tomography quantifies the fidelity of the identity gate acting on a particular qubit, offering a general assessment of the DD sequence's performance for any input state.

In this example, we consider an idle period equal to the duration of 40 π-pulses of 128 ns each without any inter-pulse delays, that is, a total idle time of $T = 5.12$ μs. One can see that UR40 delivers good performance over a very broad range of amplitude errors ϵ and frequency detunings δ, but only once the full sequence has been completed (i.e., at $t = 40\,T_\pi$). At intermediate times, its fidelity landscape appears irregular, since the pulse is not designed to refocus at shorter timescales. In contrast, the fidelity of XY4 is only high over a reasonably broad range of parameters after one or two XY4 cycles, but this range gradually shrinks as more cycles are applied, reflecting the accumulation of residual errors. As a result, by the end of the full idle time, the sequence exhibits limited robustness (concentrated around the center). U-TRACK4, by comparison, remains robust even after the full sequence of 40 π-pulses, while still benefiting from regular refocusing. The only visible difference between experiment and simulation is a global loss of fidelity due to decoherence, which the simulations do not include. We also observed similar robustness behavior when a delay is introduced between the pulses.

C. Robustness to time-dependent noise

The fidelity profile in Fig. 3 illustrates the robustness of each sequence against static deviations. It does not capture the response to time-dependent fluctuations in the Hamiltonian parameters. Such fluctuations can occur during the execution of quantum circuits due to mechanisms discussed earlier [41, 42, 50, 51, 53]. In particular, two-level fluctuators [41] or charge parity switches [41, 56], can cause the detuning to switch between two values, $\delta_{\min}$ and $\delta_{\max}$. These fluctuations can be modeled as *random telegraph noise* (RTN) [57, 58], a type of stochastic noise in which a parameter — in this case, the detuning — abruptly switches between two values at random times. The characteristics of the noise can be quantified by the switching rate, λ (in Hz), which represents the average number of times the detuning $\delta(t)$ switches per second. As a concrete example, for $T = 5.12$ μs and a switching rate of $\lambda = 1$ MHz, the expected number of switches during the idle period is $\lambda T \approx 5$, meaning the detuning switches approximately five times on average.

To evaluate the performance of a dynamical decoupling sequence under this noise, we simulate the qubit's evolution for individual realizations of the detuning $\delta(t)$, using the same Gaussian π-pulses as in the experiment (see Appendix C). For a specified value of λ, we generate a large ensemble of random detuning profiles—typically 2,000 samples. By computing the sequence fidelity for each sample, we can statistically quantify the sequence's effectiveness in suppressing the effects of RTN.

The first row of Fig. 4 displays five representative samples of time-dependent detuning to illustrate the variability of detuning with time. The second row of Fig. 4 displays the sequence fidelities for 500 such samples. The results show that UR40 performs notably worse than XY4 and U-TRACK4, with an average fidelity of $\bar{F} \simeq 0.5$ versus $\simeq 0.9$ for the other two sequences. Moreover, the fidelity of individual UR40 realizations spans nearly the full range from 0 to 1, reflecting high sample-to-sample variability, whereas XY4 and U-TRACK4 remain consistently close to unity across almost all realizations. This discrepancy arises from structural differences in the pulse sequences. Specifically, UR40 does not refocus the state until the end of

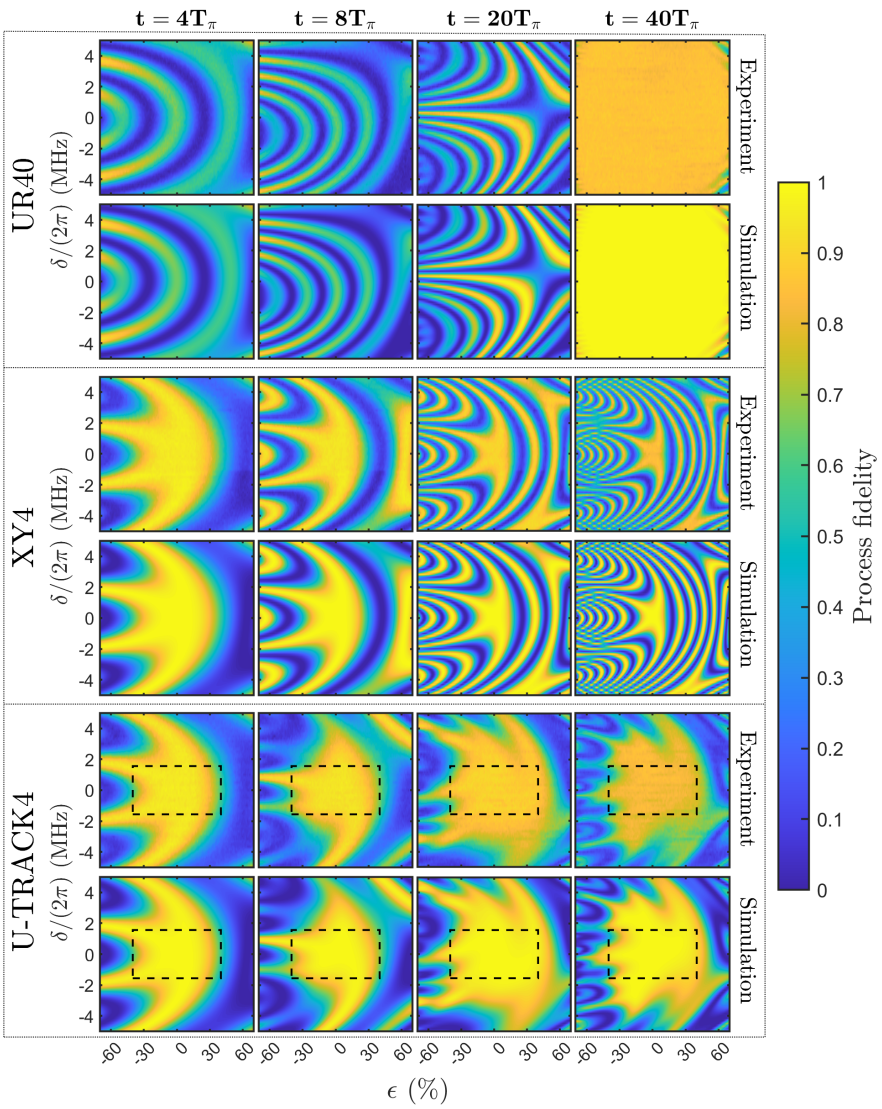


FIG. 3. Process fidelities of different dynamical decoupling sequences as functions of pulse-amplitude error ϵ and frequency detuning δ, which are intentionally engineered in the quantum device. For each sequence, the top row shows experimental measurements, while the bottom row shows the corresponding simulation results. Each column corresponds to the fidelity obtained after 4, 8, 20, and 40 π-pulses, respectively, with each π-pulse having a duration of $T_\pi = 128$ ns. The dotted rectangle shows the parameter ranges over which the U-TRACK4 sequence was optimized with assigned weights in the central region, as described in Appendix A. Technical details of the experimental implementation and simulations are reported in Appendices B and C, respectively.

the full sequence, whereas both XY4 and U-TRACK4 refocus the state after every block of four π-pulses (every 512 ns). Consequently, XY4 and U-TRACK4 exhibit reduced sensitivity to detuning fluctuations. However, this example is limited to $\lambda = 0.5$ MHz (slow fluctuations); the behavior across a broader range of noise regimes is examined in the following.

The last row of Fig. 4 gives a deeper understanding of how these sequences behave in more general scenarios. It shows the average fidelity $\bar{F}$ over 2,000 samples as a function of λ for various detuning amplitudes. Generally speaking, U-TRACK4 performs comparably to, or better than, the best of XY4 and UR40 for $\lambda \lesssim 10$ MHz, while all three sequences perform similarly for $\lambda \gtrsim 10$ MHz. Looking in more detail, four distinct regimes can be distinguished:

- High-frequency noise: $\lambda \gtrsim 10^3$ MHz (pink area).
- Medium-frequency noise: $\lambda \sim$ 1–10^3 MHz (purple area).
- Low-frequency noise: $\lambda \sim$ 0.01–1 MHz (green area).
- Static detuning: $\lambda < 0.01$ MHz (yellow area). The switching rate is so low that a switch during the idle

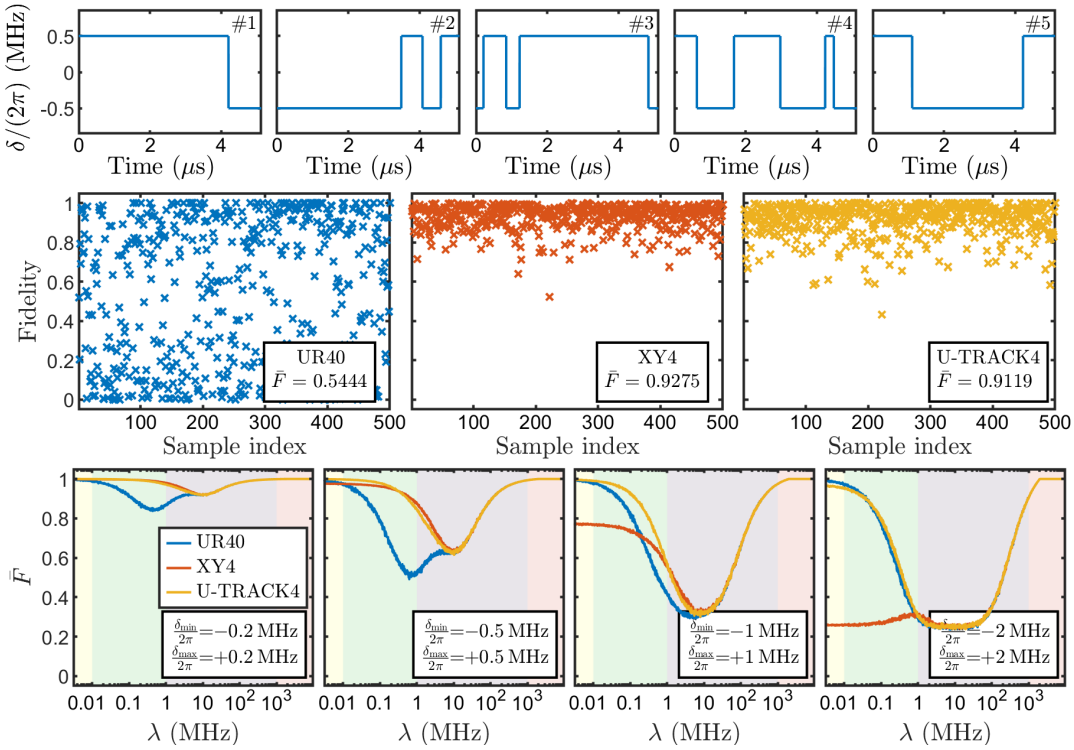


FIG. 4. *First row:* Five representative samples of detuning versus time under random telegraph noise (RTN) generated with a switching rate $\lambda = 0.5$ MHz, and detuning levels of ± 0.5 MHz, illustrating the stochastic variability of noise. *Second row:* Simulated sequence fidelity at the idle time ($T = 5.12$ μs, corresponding to 40 π-pulses) computed for 500 samples ($\lambda = 0.5$ MHz, $\delta(t)/(2\pi) = \pm 0.5$ MHz). Results are shown for the three dynamical decoupling sequences indicated in the legend. $\bar{F}$ denotes the average fidelity over the 500 samples. *Third row:* Average fidelity over 2,000 samples as a function of λ, shown for different detuning levels as indicated in the legend. The different-colored strips in the plots indicate different frequency regimes, as explained in the main text.

time is unlikely. Only two outcomes are therefore possible: $\delta = \delta_{\max}$ or $\delta = \delta_{\min}$.

For high-frequency noise (pink area), all sequences perform equally well (near $\bar{F} = 1$). In this regime, the detuning fluctuates so rapidly that its effect averages out to zero, since each sample switches between $\delta_{\max}$ and $-\delta_{\max}$.

For medium-frequency noise (purple area), all sequences perform relatively poorly. In this regime, the regular refocusing of XY4 and U-TRACK4 does not provide a clear advantage over UR40 in compensating for the fluctuations.

For low-frequency noise (green area), the differences between the three sequences become clear. In the first two panels of the last row, XY4 and U-TRACK4 maintain significantly higher average fidelity than UR40. In the second panel, for example ($\delta/(2\pi) = \pm 0.5$ MHz), UR40 drops to roughly 0.6 around $\lambda \sim 0.2$ MHz, while XY4 and U-TRACK4 remain above 0.9. In the right-most panel, XY4 performs noticeably worse, with a fidelity around 0.2 for $\lambda \lesssim 0.1$ MHz. This behavior occurs because, at such low λ, the detuning is effectively quasi-static, producing an effect similar to that discussed below.

For static detuning (yellow area), all sequences perform similarly in the first two panels, where $\delta_{\max}/(2\pi) \leq 0.5$ MHz. In this regime, the detuning is constant and relatively small, so all sequences compensate for its effect. Indeed, as shown in the last column of Fig. 3, a detuning of ± 0.5 MHz lies well within the high-fidelity region for all sequences. In contrast, a detuning of $\delta/(2\pi) = 1$ MHz is at the edge of this high-fidelity region for XY4. This explains why XY4 performs relatively poorly in the last two panels of Fig. 4 near $\lambda \sim 0.01$ MHz.

In addition, we observe that for low-frequency noise, XY4 sometimes achieves slightly higher fidelity than U-TRACK4. This is particularly visible in the second panel of the last row, for $\lambda \sim 1$ MHz. The origin of this feature is not entirely clear, but it may be related to the specific phase pattern of the sequences, which can have subtle effects. Nevertheless, the difference is very small and is not expected to have any significant impact in practice.

The poor performance of UR40 in the low-frequency

regime stems from the fragility of its fidelity profile to mid-sequence perturbations. To illustrate this, we consider a scenario in which the system parameters start at $(\epsilon_0, \delta_0) = (0, 0)$ and jump to (ϵ_1, δ_1) at $t = T/3 = 1.707\ \mu$s, analogous to the step-like profile shown in the top-left panel of Fig.4. Fig. 5 shows the simulated fidelity as a function of (ϵ_1, δ_1) at the end of the idle time; this should be compared directly with the last column of Fig. 3, which shows the static case. The key observation is that for XY4 and U-TRACK4, the high-fidelity region remains centered and well-preserved under this perturbation, whereas for UR40, the high-fidelity region is not only reduced in area but also fragmented and shifted away from the center, revealing its sensitivity to mid-sequence parameter changes.

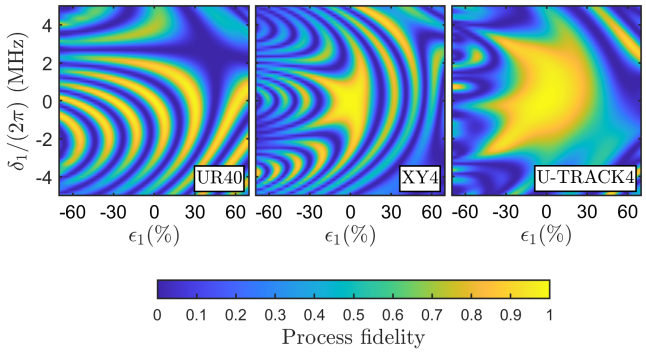


FIG. 5. Simulated fidelity of the three sequences at the end of the idle time as a function of (ϵ_1, δ_1), for a system starting at $(\epsilon_0, \delta_0) = (0, 0)$ and jumping to (ϵ_1, δ_1) at $t = T/3 = 1.707\ \mu$s, simulated with the Gaussian π-pulses described in Appendix C. Compared to the static case (last column of Fig. 3), XY4 and U-TRACK4 retain a centered, well-preserved high-fidelity region, whereas the high-fidelity region of UR40 is fragmented and shifted, revealing its sensitivity to mid-sequence parameter changes.

III. STRUCTURE OF THE U-TRACK SEQUENCES

The structure of the optimized U-TRACK sequences provides insight into the underlying control mechanisms of robust DD and helps explain why the optimization reliably converges to high-performing solutions. In this section, we demonstrate that the U-TRACK sequences generated by our optimization algorithm exhibit simple, regular phase patterns and symmetries — even though no such structure was imposed as a constraint during optimization. Specifically, even when starting from random initial pulse phases, the optimization algorithm converges to pulse blocks whose phases follow a recognizable pattern closely related to known analytical DD sequences, as illustrated in Fig. 6. To understand why this structure emerges, we first analyze the symmetries of a single pulse block, i.e., a fixed sequence of N π-pulses with

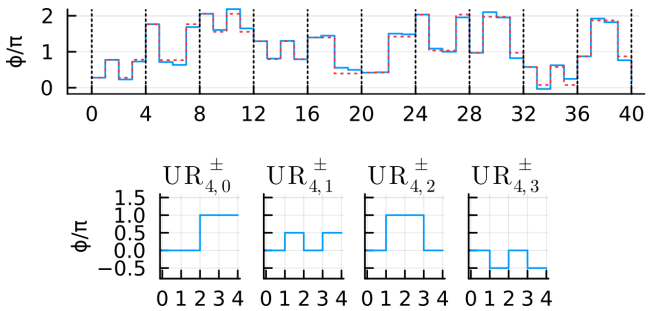


FIG. 6. Top: Optimized U-TRACK4 pulse phases for 10 blocks (solid blue). For comparison, the closest approximating XUR4 sequence (defined in the main text) is shown in red dotted. Bottom: The four distinct $\mathrm{UR}^{\pm}_{4,k}$ pulses ($k = 0, 1, 2, 3$). $\mathrm{UR}^{\pm}_{4,1}$ is equivalent to XY4 and $\mathrm{UR}^{\pm}_{4,0}$ equals the MLEV4 sequence [38, 54, 59]. The remaining two cases are obtained by applying a cyclic phase shift and possibly adding a global phase offset. Details about the pulse optimization are provided in Appendix A.

phases $(\phi_1, \ldots, \phi_N)$ that is designed to approximate the identity over one refocusing period.

(i) Global phase shift: for any phase ϕ, the sequences $(\phi_1, \ldots, \phi_N)$ and $(\phi_1+\phi, \ldots, \phi_N+\phi)$ yield the same fidelity, since the shift rotates the propagator about the z-axis (this rotation commutes with the detuning term), which leaves its trace invariant.

(ii) Cyclic permutation: The sequences $(\phi_1, \ldots, \phi_N)$ and $(\phi_2, \ldots, \phi_N, \phi_1)$ yield the same fidelity due to the cyclic property of the trace.

(iii) Reversing the pulse: The sequence $(\phi_N, \ldots, \phi_1)$ yields the same fidelity as $(\phi_1, \ldots, \phi_N)$ since $F(U^\dagger) = F(U)$ and $P(\phi, \epsilon, \delta)^\dagger = P(\phi + \pi, \epsilon, -\delta)$ (with P as in Eq. (A1)). We use the fact that the detuning range is symmetric.

(iv) Negating all phases: The sequence $(-\phi_1, \ldots, -\phi_N)$ yields the same fidelity as $(\phi_1, \ldots, \phi_N)$ since $F(\Pi_y\, U\, \Pi_y) = F(U)$ and $\Pi_y P(\phi, \epsilon, \delta)\Pi_y = P(-\phi+\pi, \epsilon, -\delta)$ where Π_y mirrors the y-axis. Again, we use the fact that the detuning range is symmetric.

Note that while these symmetries hold exactly for a single block, they will only hold approximately across multiple blocks. The single-block case is well-understood through the UR pulses introduced in [22]. For any even N greater than or equal to 4, and any $k = 0, \ldots, N-1$, one defines the sequence of phases $\phi_1, \ldots, \phi_N$ by

$$\phi_i = \frac{(i-1)(i-2)}{2}\Phi_N + (i-1)\frac{2\pi k}{N}$$

where

$$\Phi_{4m} = \pm\frac{\pi}{m}, \quad \Phi_{4m+2} = \pm\frac{2m\pi}{2m+1}\,.$$

We denote this sequence by $\mathrm{UR}^{\pm}_{N,k}$. Note that in this definition $\phi_1 = 0$, but equivalent pulses can be obtained by adding an arbitrary overall phase.

The family of UR pulses is closed under the symmetry operations listed above: If $\mathcal{S}((\phi_1, \ldots, \phi_N)) = (\phi_2, \ldots, \phi_N, \phi_1)$ is the cyclic left shift, then

$$\mathcal{S}(\mathrm{UR}^{\pm}_{N,k}) = \frac{2\pi k}{N} + \mathrm{UR}^{\pm}_{N,k\pm s}$$

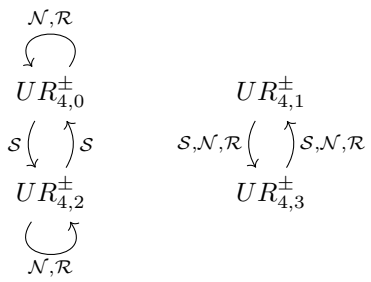
where $s = 2$ if $N = 0 \mod 4$ and $s = \frac{N}{2} - 1$ if $N = 2 \mod 4$. If $\mathcal{N}((\phi_1, \ldots, \phi_N)) = (-\phi_1, \ldots, -\phi_N)$ flips all phases, then

$$\mathcal{N}(\mathrm{UR}^{\pm}_{N,k}) = \mathrm{UR}^{\mp}_{N,N-k}\,.$$

Finally if $\mathcal{R}((\phi_1, \ldots, \phi_N)) = (\phi_N, \ldots, \phi_1)$ reverses all the phases then

$$\mathcal{R}(\mathrm{UR}^{\pm}_{N,k}) = \frac{2\pi(\pm s - k)}{N} + \mathrm{UR}^{\pm}_{N,\pm 2s-k}\,.$$

In the case of UR4, the following diagram summarizes how the four pulses $\mathrm{UR}^{\pm}_{4,k}$ $(k = 0, 1, 2, 3)$ are related to each other via the symmetry operations $\mathcal{S}$ (cyclic shift), $\mathcal{N}$ (phase negation), and $\mathcal{R}$ (pulse reversal). This can be verified by inspecting the bottom panel of Fig. 6: for example, $\mathrm{UR}^{\pm}_{4,2}$ can be obtained from $\mathrm{UR}^{\pm}_{4,0}$ (MLEV4) by applying a cyclic shift to the phases, as reflected in the step pattern of the two panels.



While UR blocks are typically repeated periodically to obtain longer sequences, periodically repeated pulses tend to accumulate errors over time, as discussed in Sec. II B. More robust sequences can be obtained by combining different UR blocks of equal length — we call pulses of this general form XUR pulses. Indeed, the GRAPE-optimized U-TRACK pulses tend to take the form of XUR pulses (Fig. 6), i.e., a structured mix of different $\mathrm{UR}^{\pm}_{4,k}$ blocks rather than a periodic repetition of a single block. As a result, they outperform periodically repeated UR pulses in robustness (Fig. 3). Furthermore, initializing the GRAPE optimization from an XUR ansatz rather than a random initial guess leads to faster convergence, confirming that XUR pulses provide a physically motivated and effective starting point for numerical optimization.

This behavior closely resembles the fourth approach briefly mentioned in Sec. II A, which consists of applying short DD sequences with cyclic shifts or sign changes between blocks [8, 9, 46, 54]. Indeed, for this specific example, XY64 achieves similar performance to U-TRACK4, as shown in more detail in Appendix D. More generally, when using blocks of 4 π-pulses, U-TRACK reproduces the performance of XY64 or MLEV-type sequences; when using a single block spanning the full idle time, it recovers performance comparable to the UR family. Notably, the optimized U-TRACK sequences do not simply recover known supercycles: the XUR structure found by our optimization scheme mixes MLEV-type and XY-type blocks within a single sequence — a combination that, to our knowledge, has not been considered in the literature, where MLEV-type and XY-type supercycles have been studied separately [8, 9, 46].

IV. DISCUSSION AND OUTLOOK

In this work, we introduced U-TRACK, a new family of dynamical decoupling sequences derived from an optimal tracking strategy. The tracking strategy monitors the qubit's propagator after each pulse block during optimization, enabling dynamic error compensation that suppresses error accumulation block by block. Conventional periodic DD schemes repeat a fixed block, which often leads to systematic error accumulation across repetitions. U-TRACK instead compensates residual errors dynamically at every block, yielding non-periodic pulse schedules that remain robust over long idle times. To efficiently search the space of pulse parameters, we combine GRAPE-based gradient optimization with a genetic algorithm: the genetic algorithm explores the parameter space globally and provides diverse initial guesses, while GRAPE refines them locally to high-fidelity solutions. Furthermore, the optimized sequences exhibit a structured symmetry that can be exploited to warm-start the optimization: using a symmetry-informed initial guess rather than a random one significantly reduces convergence time and improves solution quality. U-TRACK is a general framework: it supports an arbitrary number of blocks M and any even number of pulses N per block, making it applicable across a wide range of hardware constraints and idle-time durations. In this work, we focus on the case $N = 4$ as a concrete demonstration, but the framework extends directly to other configurations.

We demonstrated the effectiveness of this framework through the U-TRACK4 sequence, validating its robustness both numerically and experimentally. Experimental validation was performed on a transmon processor at the Walther-Meißner-Institut (WMI), while additional experiments on IQM devices are presented below, demonstrating the cross-platform applicability of U-TRACK sequences. These WMI experiments show substantial improvements in resilience to both static detuning and pulse-amplitude errors. U-TRACK4

maintains high process fidelity over long idle times, whereas XY4 suffers from cumulative residual errors. Simulations further show that UR40 remains sensitive to mid-sequence parameter changes (Sec. II C). Under random telegraph noise (Sec. II C), the three sequences perform similarly in the high-frequency regime, where rapid fluctuations average out regardless of sequence structure. In the low-frequency regime and for small detuning amplitudes, however, U-TRACK4 and XY4 maintain significantly higher fidelities than UR40, whose performance degrades severely when mid-sequence parameter changes occur. Within this low-frequency regime, U-TRACK4 further outperforms XY4: when the effective detuning is relatively high, XY4 exhibits sharp fidelity degradation that U-TRACK4 avoids. In addition to these controlled robustness experiments, we performed a dedicated comparison of U-TRACK4 and XY4 on the IQM Garnet device [60] under native operating conditions, as detailed in Appendix E. These measurements confirm that U-TRACK4 maintains a clear advantage over XY4 in regimes dominated by pulse distortions and overlap, while providing comparable protection once modest interpulse delays are introduced to mitigate such effects (see Fig. 10).

It is important to acknowledge that established sequences such as XY64 and MLEV-type supercycles [8, 9, 46], which combine cyclic and sign-reversed variants of short base blocks, can achieve performance comparable to U-TRACK4 when applied to sequences of four π-pulses. This connection is not coincidental: as the analysis in Sec. III and Appendix D demonstrates, the optimal tracking algorithm naturally converges to a combination of MLEV-type and XY-type blocks — a structure we term XUR. Notably, to our knowledge, this mixed structure has not been considered in the prior literature, where MLEV-type and XY-type supercycles have been studied separately [8, 9, 46]. The XUR structure also has practical implications for optimization: using an XUR sequence as a warm-start initial guess leads to faster convergence and improved solution quality compared to random initialization, as discussed in Sec. III.

The U-TRACK framework nevertheless offers several key advantages: (i) it is not limited to π-gates: the current implementation optimizes phases of π-rotations, but general pulse-phase shaping could provide additional degrees of freedom to improve robustness and reduce the refocusing period; (ii) it allows flexible control over the number of pulses per block (e.g., U-TRACK4, U-TRACK6, U-TRACK8), whereas XY64 and MLEV sequences are both limited to blocks of 4 pulses and fixed to specific phase patterns; (iii) the optimization can incorporate arbitrary noise models, enabling sequences tailored to device-specific noise spectra; and (iv) the robustness profile can be designed to target specific spectral regions or parameter ranges, as highlighted in previous studies [61, 62].

Looking forward, several promising directions emerge for extending the U-TRACK framework. The tracking strategy could be generalized beyond single-qubit settings to multi-qubit architectures, where correlated noise, crosstalk, and gate-interaction idling introduce new challenges [50, 63]. A natural starting point is the combinatorial and group-based decoupling constructions developed for interacting spin systems [40], which could serve as building blocks for tracking-based optimization over multi-qubit blocks. Extending U-TRACK to protect entangled states or idle qubits within larger subroutines may further enhance the stability of noisy quantum processors. In addition, incorporating more advanced pulse-shaping capabilities may enable simultaneous suppression of multiple noise channels, including faster time-varying or spectrally complex noise. Dynamically tailored pulse envelopes, durations, and frequencies—beyond the π-pulse framework used here—could further expand the practical operating regime of the sequences.

The optimization toolkit is released as an open-source Julia package, UTrack.jl [47], facilitating integration into quantum-control pipelines and experimental workflows. As quantum hardware scales, such dynamically optimized and hardware-agnostic DD sequences may become an essential component of high-fidelity quantum computation.

V. CODE AVAILABILITY

The optimization toolkit developed in this work is released as an open-source Julia package, UTrack.jl [47], which supports both CPU and GPU execution for flexible deployment across different computational platforms. The repository includes ready-to-use, pre-optimized U-TRACK sequences with varying pulse-block lengths.

VI. ACKNOWLEDGEMENTS

The project is part of the Munich Quantum Valley, which is supported by the Bavarian state government with funds from the Hightech Agenda Bayern Plus. The authors acknowledge financial support from the German Federal Ministry of Education and Research (BMBF) through the programs "German Quantum Computer based on Superconducting Qubits" (GeQCoS; No. 13N15680) and MUNIQCSC (No. 13N16188), and from the Deutsche Forschungsgemeinschaft (DFG) through the excellence initiative "Munich Center for Quantum Science and Technology" (MCQST; No. 390814868). We thank Niklas Bruckmoser and Julius Feigl for fabricating the superconducting qubit processors and Ivan Tsitsilin

for the QPU design used at WMI. AA and IR acknowledge the support of the UK government Department for Science, Innovation and Technology (DSIT) through the UK National Quantum Technologies Programme and the International Science Partnership Fund (ISPF)

[1] P. W. Shor, Scheme for reducing decoherence in quantum computer memory, Physical review A **52**, R2493 (1995).

[2] J. Preskill, Quantum computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

[3] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, A quantum engineer's guide to superconducting qubits, Applied physics reviews **6** (2019).

[4] Google Quantum AI and Collaborators, Quantum error correction below the surface code threshold, Nature **638**, 920 (2025).

[5] E. L. Hahn, Spin echoes, Physical Review **80**, 580 (1950).

[6] H. Y. Carr and E. M. Purcell, Effects of diffusion on free precession in nuclear magnetic resonance experiments, Physical review **94**, 630 (1954).

[7] S. Meiboom and D. Gill, Modified spin-echo method for measuring nuclear relaxation times, Review of Scientific Instruments **29**, 688 (1958).

[8] M. H. Levitt, R. Freeman, and T. Frenkiel, Supercycles for broadband heteronuclear decoupling, Journal of Magnetic Resonance (1969) **50**, 157 (1982).

[9] T. Gullion, D. B. Baker, and M. S. Conradi, New, compensated carr-purcell sequences, Journal of Magnetic Resonance (1969) **89**, 479 (1990).

[10] A. M. Souza, G. A. Alvarez, and D. Suter, Robust dynamical decoupling for quantum computing and quantum memory, Physical Review Letters **106**, 240501 (2011).

[11] J. Bylander, S. Gustavsson, F. Yan, F. Yoshihara, K. Harrabi, G. Fitch, D. G. Cory, Y. Nakamura, J.-S. Tsai, and W. D. Oliver, Noise spectroscopy through dynamical decoupling with a superconducting flux qubit, Nature Physics **7**, 565 (2011).

[12] N. Ezzell, B. Pokharel, L. Tewala, G. Quiroz, and D. A. Lidar, Dynamical decoupling for superconducting qubits: A performance survey, Physical Review Applied **20**, 064027 (2023).

[13] S. Gustavsson, F. Yan, J. Bylander, F. Yoshihara, Y. Nakamura, T. P. Orlando, and W. D. Oliver, Dynamical decoupling and dephasing in interacting two-level systems, Physical review letters **109**, 010502 (2012).

[14] J.-X. Han, J. Zhang, G.-M. Xue, H. Yu, and G. Long, Protecting logical qubits with dynamical decoupling, Physical Review Applied **24**, 024003 (2025).

[15] M. J. Biercuk, H. Uys, A. P. VanDevender, N. Shiga, W. M. Itano, and J. J. Bollinger, Optimized dynamical decoupling in a model quantum memory, Nature **458**, 996 (2009).

[16] M. J. Biercuk, H. Uys, A. P. VanDevender, N. Shiga, W. M. Itano, and J. J. Bollinger, Experimental uhrig dynamical decoupling using trapped ions, Physical Review A—Atomic, Molecular, and Optical Physics **79**, 062324 (2009).

[17] Y. Sagi, I. Almog, and N. Davidson, Process tomography of dynamical decoupling in a dense cold atomic ensemble, Physical Review Letters **105**, 053201 (2010).

[18] S. Damodarakurup, M. Lucamarini, G. Di Giuseppe, D. Vitali, and P. Tombesi, Experimental inhibition of decoherence on flying qubits via "bang-bang" control, Physical Review Letters **103**, 040502 (2009).

[19] C. L. Degen, F. Reinhard, and P. Cappellaro, Quantum sensing, Reviews of Modern Physics **89**, 035002 (2017).

[20] Z.-H. Wang, G. De Lange, D. Ristè, R. Hanson, and V. Dobrovitski, Comparison of dynamical decoupling protocols for a nitrogen-vacancy center in diamond, Physical Review B—Condensed Matter and Materials Physics **85**, 155204 (2012).

[21] A. Maudsley, Modified carr-purcell-meiboom-gill sequence for NMR fourier imaging applications, Journal of Magnetic Resonance (1969) **69**, 488 (1986).

[22] G. T. Genov, D. Schraft, N. V. Vitanov, and T. Halfmann, Arbitrarily accurate pulse sequences for robust dynamical decoupling, Physical Review Letters **118**, 133202 (2017).

[23] B. Pokharel and D. A. Lidar, Better-than-classical Grover search via quantum error detection and suppression, npj Quantum Information **10**, 23 (2024).

[24] P. Singkanipa, V. Kasatkin, Z. Zhou, G. Quiroz, and D. A. Lidar, Demonstration of algorithmic quantum speedup for an abelian hidden subgroup problem, Physical Review X **15**, 021082 (2025).

[25] B. Pokharel and D. A. Lidar, Demonstration of algorithmic quantum speedup, Physical Review Letters **130**, 210602 (2023).

[26] C. Tong, H. Zhang, and B. Pokharel, Empirical learning of dynamical decoupling on quantum processors, PRX Quantum **6**, 030319 (2025).

[27] A. Rahman, D. J. Egger, and C. Arenz, Learning how to dynamically decouple by optimizing rotational gates, Physical Review Applied **22**, 054074 (2024).

[28] Qiskit Community, Qiskit: An open-source framework for quantum computing (2017), accessed: 2026-02.

[29] A. M. Souza, G. A. Álvarez, and D. Suter, Robust dynamical decoupling, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **370**, 4748 (2012).

[30] D. Suter and G. A. Álvarez, Colloquium: Protecting quantum information against environmental noise, Reviews of Modern Physics **88**, 041001 (2016).

[31] M. H. Levitt and R. Freeman, Nmr population inversion using a composite pulse, Journal of Magnetic Resonance (1969) **33**, 473 (1979).

[32] M. H. Levitt, Composite pulses, Progress in Nuclear Magnetic Resonance Spectroscopy **18**, 61 (1986).

[33] R. Freeman, *Spin choreography* (Oxford University Press Oxford, 1998).

[34] K. Khodjasteh and D. A. Lidar, Fault-tolerant quantum dynamical decoupling, Physical Review Letters **95**, 180501 (2005).

[35] K. Khodjasteh and D. A. Lidar, Performance of deterministic dynamical decoupling schemes: Concatenated and periodic pulse sequences, Physical Review

A—Atomic, Molecular, and Optical Physics **75**, 062310 (2007).
[36] K. Kobzar, S. Ehni, T. E. Skinner, S. J. Glaser, and B. Luy, Exploring the limits of broadband 90° and 180° universal rotation pulses, Journal of Magnetic Resonance **225**, 142 (2012).
[37] T. E. Skinner, N. I. Gershenzon, M. Nimbalkar, W. Bermel, B. Luy, and S. J. Glaser, New strategies for designing robust universal rotation pulses: Application to broadband refocusing at low power, Journal of Magnetic Resonance **216**, 78 (2012).
[38] M. H. Levitt and R. Freeman, Compensation for pulse imperfections in NMR spin-echo experiments, Journal of Magnetic Resonance (1969) **43**, 65 (1981).
[39] L. F. Santos and L. Viola, Enhanced convergence and robust performance of randomized dynamical decoupling, Phys. Rev. Lett. **97**, 150501 (2006).
[40] L. F. Santos and L. Viola, Advantages of randomization in coherent quantum dynamical control, New Journal of Physics **10**, 083009 (2008).
[41] A. Agarwal, L. P. Lindoy, D. Lall, S. E. de Graaf, T. Lindström, and I. Rungger, Fast-tracking and disentangling of qubit noise fluctuations using minimal-data averaging and hierarchical discrete fluctuation auto-segmentation (2025), arXiv:2505.23622 [quant-ph].
[42] J. M. Martinis, K. B. Cooper, R. McDermott, M. Steffen, M. Ansmann, K. D. Osborn, K. Cicak, S. Oh, D. P. Pappas, R. W. Simmonds, and C. C. Yu, Decoherence in josephson qubits from dielectric loss, Phys. Rev. Lett. **95**, 210503 (2005).
[43] J. L. Neves, B. Heitmann, N. Khaneja, and S. J. Glaser, Heteronuclear decoupling by optimal tracking, Journal of Magnetic Resonance **201**, 7 (2009).
[44] F. Schilling, L. R. Warner, N. I. Gershenzon, T. E. Skinner, M. Sattler, and S. J. Glaser, Next-generation heteronuclear decoupling for high-field biomolecular NMR spectroscopy, Angewandte Chemie International Edition **53**, 4475 (2014).
[45] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms, Journal of Magnetic Resonance **172**, 296 (2005).
[46] R. Freeman, T. Frenkiel, and M. H. Levitt, A simple "black-box" decoupler, Journal of Magnetic Resonance (1969) **50**, 345 (1982).
[47] E. Malvetti, L. V. Damme, A. Devra, and S. J. Glaser, UTrack.jl (2026).
[48] S. Sheldon, L. S. Bishop, E. Magesan, S. Filipp, J. M. Chow, and J. M. Gambetta, Characterizing errors on qubit operations via iterative randomized benchmarking, Phys. Rev. A **93**, 012301 (2016).
[49] E. Wright, L. Van Damme, N. Glaser, A. Devra, F. Roy, J. Englhardt, N. Bruckmoser, L. Koch, A. Marx, J. Schirk, C. Schneider, L. Södergren, I. Tsitsilin, F. Wallner, S. Glaser, M. Werninghaus, and S. Filipp, Superconducting-qubit gates robust to parameter fluctuations, Phys. Rev. Appl. **25**, 054037 (2026).
[50] V. Tripathi, H. Chen, M. Khezri, K.-W. Yip, E. Levenson-Falk, and D. A. Lidar, Suppression of Crosstalk in Superconducting Qubits Using Dynamical Decoupling, Phys. Rev. Appl. **18**, 024068 (2022), publisher: American Physical Society.
[51] A. Agarwal, L. P. Lindoy, D. Lall, F. Jamet, and I. Rungger, Modelling non-markovian noise in driven superconducting qubits, QST **9**, 035017 (2024).
[52] P. Kumar, S. Sendelbach, M. A. Beck, J. W. Freeland, Z. Wang, H. Wang, C. C. Yu, R. Q. Wu, D. P. Pappas, and R. McDermott, Origin and reduction of $1/f$ magnetic flux noise in superconducting devices, Phys. Rev. Appl. **6**, 041001 (2016).
[53] E. Hyyppä, A. Vepsäläinen, M. Papič, C. F. Chan, S. Inel, A. Landra, W. Liu, J. Luus, F. Marxer, C. Ockeloen-Korppi, S. Orbell, B. Tarasinski, and J. Heinsoo, Reducing leakage of single-qubit gates for superconducting quantum processors using analytical control pulse envelopes, PRX Quantum **5**, 030353 (2024).
[54] D. M. Lancaster, M. A. Shahbaz, H. G. Yousefabad, S. Chatterjee, E. Ram, and J. D. Weinstein, Quantum sensing in the presence of pulse errors and qubit leakage (2025), arXiv:2509.09874 [quant-ph].
[55] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, 2010).
[56] D. M. Tennant, L. A. Martinez, K. M. Beck, S. R. O'Kelley, C. D. Wilen, R. McDermott, J. L. DuBois, and Y. J. Rosen, Low-frequency correlated charge-noise measurements across multiple energy transitions in a tantalum transmon, PRX Quantum **3**, 030307 (2022).
[57] E. Paladino, Y. M. Galperin, G. Falci, and B. L. Altshuler, $1/f$ noise: Implications for solid-state quantum information, Rev. Mod. Phys. **86**, 361 (2014).
[58] L. Cywiński, R. M. Lutchyn, C. P. Nave, and S. Das Sarma, How to enhance dephasing time in superconducting qubits, Phys. Rev. B **77**, 174509 (2008).
[59] M. H. Levitt, R. Freeman, and T. Frenkiel, Broadband heteronuclear decoupling, Journal of Magnetic Resonance (1969) **47**, 328 (1982).
[60] L. Abdurakhimov, J. Adam, H. Ahmad, O. Ahonen, *et al.*, Technology and performance benchmarks of IQM's 20-qubit quantum computer (2024), arXiv:2408.12433 [quant-ph].
[61] K. Kobzar, B. Luy, N. Khaneja, and S. J. Glaser, Pattern pulses: design of arbitrary excitation profiles as a function of pulse amplitude and offset, Journal of Magnetic Resonance **173**, 229 (2005).
[62] M. A. Janich, R. F. Schulte, M. Schwaiger, and S. J. Glaser, Robust slice-selective broadband refocusing pulses, Journal of Magnetic Resonance **213**, 126 (2011).
[63] E. Hickman, X. Wu, and G. Quiroz, Crosstalk-robust dynamical decoupling for bipartite-topology quantum processors, Phys. Rev. Appl. **25**, 064041 (2026).
[64] K. Petersen and M. Pedersen, *The Matrix Cookbook* (Technical University of Denmark, 2006) version 20051003.
[65] A. Devra, P. Prabhu, H. Singh, Arvind, and K. Dorai, Efficient experimental design of high-fidelity three-qubit quantum gates via genetic programming, Quantum Information Processing **17**, 67 (2018).
[66] G. Quiroz and D. A. Lidar, Optimized dynamical decoupling via genetic algorithms, Phys. Rev. A **88**, 052306 (2013).
[67] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, Charge-insensitive qubit design derived from the cooper pair box, Phys. Rev. A **76**, 042319 (2007).

[68] N. Glaser, F. Roy, I. Tsitsilin, L. Koch, N. Bruckmoser, J. Schirk, J. Romeiro, G. Huber, F. Wallner, M. Singh, G. Krylov, A. Marx, L. Södergren, C. Schneider, M. Werninghaus, and S. Filipp, Closed-loop optimization for high-fidelity controlled-z gates in superconducting qubits, Phys. Rev. Appl. **24**, 024048 (2025).

[69] Zurich Instruments AG, *SHFSG+ User Manual: 8.5 GHz Signal Generator*, Zurich Instruments AG, Technoparkstrasse 1, 8005 Zurich, Switzerland, revision 26.04 ed. (2026), accessed: 2026-06-17.

[70] N. Johnston and D. W. Kribs, Quantum gate fidelity in terms of choi matrices, Journal of Physics A: Mathematical and Theoretical **44**, 495303 (2011).

Appendix A: Optimal tracking sequences

1. Optimization of U-TRACK sequences

a. Overview

The sequence is optimized using a composite pulse made of a concatenation of several π-rotations with numerically optimized phases ϕ_n without inter-pulse delays. One π-rotation of phase ϕ_n is assumed to be of the form:

$$P_n^{\epsilon\delta} = e^{-i\phi_n \frac{\sigma_z}{2}} e^{-i\left(\Omega(1+\epsilon)\frac{\sigma_x}{2} + \delta\frac{\sigma_z}{2}\right)T_\pi} e^{i\phi_n \frac{\sigma_z}{2}}, \tag{A1}$$

where $\sigma_{x,y,z}$ are the Pauli matrices, Ω is the Rabi rate in rad/s, $T_\pi = \pi/\Omega$ is the duration of a π-pulse, ϕ_n is the phase of the n-th pulse, and where δ and ϵ represent the effects of detuning and pulse amplitude deviations, respectively. After the first DD block, the evolution operator becomes:

$$U_1^{\epsilon\delta} = P_N^{\epsilon\delta} P_{N-1}^{\epsilon\delta} \cdots P_1^{\epsilon\delta}, \tag{A2}$$

where N is the number of π-pulses in one block. After the second block, the evolution operator is:

$$U_2^{\epsilon\delta} = P_{2N}^{\epsilon\delta} P_{2N-1}^{\epsilon\delta} \cdots P_{N+1}^{\epsilon\delta} U_1^{\epsilon\delta}, \tag{A3}$$

After M blocks, the operator is:

$$U_M^{\epsilon\delta} = P_{M\cdot N}^{\epsilon\delta} P_{M\cdot N-1}^{\epsilon\delta} \cdots P_{M\cdot N-N+1}^{\epsilon\delta} U_{M-1}^{\epsilon\delta}, \tag{A4}$$

which is made of $M \times N$ π-pulses of phases ϕ_{mn}.

In a standard DD sequence, such as XY4, the pulse phases $\phi_1, \ldots, \phi_N$ are optimized so that, after a single block, the resulting evolution operator satisfies $U^{(1)} \simeq \mathbb{I}$ over a given region of parameter space (ϵ, δ). Specifically, this can be achieved by optimizing $\phi_1, \ldots, \phi_N$ to maximize the fidelity:

$$F_1 = \eta \int_{\delta_{\min}}^{\delta_{\max}} \int_{\epsilon_{\min}}^{\epsilon_{\max}} \frac{1}{4} \Big| \operatorname{Tr}\left(\mathbb{I} \cdot U_1(\delta, \epsilon)\right) \Big|^2 \, d\epsilon \, d\delta, \tag{A5}$$

where $\eta = 1/[(\epsilon_{\max} - \epsilon_{\min})(\delta_{\max} - \delta_{\min})]$ normalizes the fidelity to the area of the parameter region.

The optimized block is subsequently repeated multiple times to span the qubit's idle period. Consequently, after M blocks, the evolution operator is given by $U_M = U_1^M$. This implies that the fidelities F_2, F_3, and so forth are fully determined by the first block, and that any residual errors present in this block accumulate over successive repetitions.

In our approach, all blocks are optimized simultaneously by finding a set of pulse phases $\phi_1, \cdots, \phi_{M\times N}$ that maximizes the total fidelity

$$F = \frac{1}{M} \sum_{m=1}^{M} F_m \tag{A6}$$

with:

$$F_m = \eta_w \int_{\delta_{\min}}^{\delta_{\max}} \int_{\epsilon_{\min}}^{\epsilon_{\max}} w(\epsilon,\delta)\,\frac{1}{4}\Big|\,\mathrm{Tr}\left(\mathbb{I}\cdot U_m(\delta,\epsilon)\right)\Big|^2\,d\epsilon\,d\delta, \tag{A7}$$

where the weight $w(\epsilon,\delta)$ and its normalization η_w are given below.

With this total fidelity being maximized, if residual errors remain after the first block, the subsequent blocks adapt their pulse phases to compensate for them, and this correction process continues throughout the sequence. This approach maintains a high fidelity across the parameter space even for long idle durations.

The U-TRACK4 sequence presented in the main text was optimized over $M = 100$ blocks of $N = 4$ π-pulses, with parameter ranges $\delta/(2\pi) \in [-1.5625, 1.5625]$ MHz and $\epsilon \in [-0.4, 0.4]$. The maximum detuning corresponds to 0.4 times the Rabi frequency of a π-pulse of duration $T_\pi = 128$ ns. Additional weight was assigned to the central region of the parameter space ($\epsilon \sim 0$, $\delta \sim 0$) through a Gaussian factor of the form:

$$w(\epsilon,\delta) = 1 + w_0 \exp\left\{-\frac{1}{2\sigma^2}\left[\left(\tfrac{\epsilon}{\epsilon_{\max}}\right)^2 + \left(\tfrac{\delta}{\delta_{\max}}\right)^2\right]\right\}$$

with $w_0 = 100$ and $\sigma = 0.3$. Accordingly, the normalization in Eq. (A7) is $\eta_w = 1/\iint w(\epsilon,\delta)\,d\epsilon\,d\delta$, which ensures $F_m \leq 1$.

A careful reader may note that we could optimize the fidelity of the form

$$F_m^{\pm} = \pm\eta_w \int_{\delta_{\min}}^{\delta_{\max}} \int_{\epsilon_{\min}}^{\epsilon_{\max}} w(\epsilon,\delta)\,\frac{1}{2}\,\mathbb{R}[\mathrm{Tr}(\mathbb{I}\cdot U_m(\delta,\epsilon))]\,d\epsilon\,d\delta, \tag{A8}$$

where the sign is chosen by the user, instead of Eq. (A7). The difference is crucial from an optimal-control perspective, as explained below. In Eq. (A7), the sign of the identity is not imposed; that is, both $U = -\mathbb{I}$ and $U = +\mathbb{I}$ yield perfect fidelity. By contrast, in Eq. (A8) the sign is fixed; for instance, for F^+ the fidelity is poor when $U = -\mathbb{I}$. Nevertheless, previous work has shown that the type fidelity given in Eq. (A8) converges much more reliably toward the global optimum [36], although it is not clear under what circumstances it is preferable to impose the + or − sign. In our case, since there are many fidelity subfunctions F_m, Eq. (A8) is not appropriate, as the best choice of sign for each F_m is unclear. It would require exploring two possible choices for each F_m, resulting in 2^M possible combinations of signs to identify the global optimum.

To optimize the total fidelity in Eq. (A6), we use a modified version of the GRAPE algorithm [45], a gradient-based method that converges to local optimal solutions near the initial pulse guess. Technical details of the algorithm are provided in Appendix A 2 below, and an open source package is available online [47]. We observed that the solutions produced by GRAPE strongly depend on the initial pulse guess. This sensitivity becomes more pronounced when optimizing over a larger number of blocks M , suggesting a control landscape with many competing local minima. Finding the best solutions requires optimizing with numerous initial pulse guesses, which can be cumbersome since a single optimization run typically takes a few dozen minutes on a standard laptop. To improve the efficiency of the optimization, we enhanced GRAPE with a global search based on a genetic algorithm (see Sec. A 3), which substantially increases the likelihood of obtaining the global optimum, or at least high-quality locally-optimal solutions.

2. Local optimization via gradient descent

Propagation. As explained earlier, the optimization is performed assuming an ensemble of two-level systems of different amplitude errors ϵ and detuning δ. For a given pair (ϵ,δ), the evolution operator after M DD blocks is given by:

$$\underbrace{\underbrace{P_{MN}^{\epsilon\delta}P_{MN-1}^{\epsilon\delta}\cdots P_{2N+1}^{\epsilon\delta}}_{U_2^{\epsilon\delta}}\underbrace{P_{2N}^{\epsilon\delta}\cdots P_{N+1}^{\epsilon\delta}}_{U_2^{\epsilon\delta}}\underbrace{P_{N}^{\epsilon\delta}\cdots P_{1}^{\epsilon\delta}}_{U_1^{\epsilon\delta}}}_{U_M^{\epsilon\delta}}, \tag{A9}$$

Optimized fidelity. In practice, the error parameters ϵ and δ are discretized in a grid of $N_\epsilon \times N_\delta$ pairs of values $(\epsilon_k, \delta_\ell)$, distributed such that $\epsilon_k \in [-\epsilon_{\max}, \epsilon_{\max}]$ and $\delta_\ell \in [-\delta_{\max}, \delta_{\max}]$. The fidelity of the sequence after m blocks—given by Eq. (A7)—is evaluated using:

$$\begin{aligned} F_m &= \frac{1}{W}\sum_{k=1}^{N_\epsilon}\sum_{\ell=1}^{N_\delta} w_{k\ell}\,\frac{1}{4}\Big|\,\mathrm{Tr}\left(\mathbb{I}\cdot U_m^{\epsilon_k\delta_\ell}\right)\Big|^2 \\ &\equiv \frac{1}{W}\sum_{k=1}^{N_\epsilon}\sum_{\ell=1}^{N_\delta} w_{k\ell}\,f_m^{\epsilon_k\delta_\ell}, \end{aligned} \tag{A10}$$

representing the weighted average fidelity on the grid, with $w_{k\ell} = w(\epsilon_k,\delta_\ell)$ and $W = \sum_{k,\ell} w_{k\ell}$. The total fidelity that is optimized using optimal tracking is given by:

$$F = \frac{1}{M}\sum_{m=1}^{M} F_m, \tag{A11}$$

which corresponds to the average fidelity over all blocks. The goal is thus to find a set of pulse phases ϕ_n that maximizes F. To do so, we use a gradient algorithm in which gradients are calculated using GRAPE, as explained below.

Gradients. The gradients correspond to the derivatives of the fidelity with respect to all $M \times N$ phases. They are given by:

$$\frac{\partial F}{\partial \phi_n} = \frac{1}{M}\sum_{m=1}^{M}\frac{\partial F_m}{\partial \phi_n}. \tag{A12}$$

Note that since F_m is the fidelity after m blocks, we obtain $\partial F_m/\partial\phi_n = 0$ if $n > m \times N$ (N is the number of pulses in one block). This means that the above formula can be rewritten as:

$$\frac{\partial F}{\partial \phi_n} = \frac{1}{M} \sum_{m \geq \frac{n}{N}}^{M} \frac{\partial F_m}{\partial \phi_n}. \tag{A13}$$

Continuing to apply the chain rule, we get:

$$\frac{\partial F}{\partial \phi_n} = \frac{1}{MW} \sum_{m \geq \frac{n}{N}}^{M} \sum_{k=1}^{N_\epsilon} \sum_{\ell=1}^{N_\delta} w_{k\ell} \frac{\partial f_m^{\epsilon_k \delta_\ell}}{\partial \phi_n}. \tag{A14}$$

Since $f_m^{\epsilon_k\delta_\ell}$ is determined by the evolution operators $U_m^{\epsilon_k\delta_\ell}$, we can use the chain rule [64]:

$$\frac{\partial f_m^{\epsilon_k \delta_\ell}}{\partial \phi_n} = \mathbb{R}\left[\mathrm{Tr}\left(\frac{\partial f_m^{\epsilon_k \delta_\ell}}{\partial U_m^{\epsilon_k \delta_\ell}} \frac{\partial U_m^{\epsilon_k \delta_\ell}}{\partial \phi_n}\right)\right]. \tag{A15}$$

The first term is a derivative of a scalar with respect to a matrix, leading to:

$$\frac{\partial f_m^{\epsilon_k \delta_\ell}}{\partial U_m^{\epsilon_k \delta_\ell}} = \frac{1}{2} \mathrm{Tr}\left(U_m^{\epsilon_k \delta_\ell}\right)^* \mathbb{I}. \tag{A16}$$

The second term is given by:

$$\frac{\partial U_m^{\epsilon_k \delta_\ell}}{\partial \phi_n} = P_{mN}^{\epsilon_k \delta_\ell} \cdots P_{n+1}^{\epsilon_k \delta_\ell} \frac{\partial P_n^{\epsilon_k \delta_\ell}}{\partial \phi_n} P_{n-1}^{\epsilon_k \delta_\ell} \cdots P_1^{\epsilon_k \delta_\ell}. \tag{A17}$$

The term in the middle is derived using Eq. (A1), which leads to:

$$\frac{\partial P_n^{\epsilon_k \delta_\ell}}{\partial \phi_n} = -i\left[\frac{\sigma_z}{2}, P_n^{\epsilon_k \delta_\ell}\right], \tag{A18}$$

where $[\cdot,\cdot]$ represents the Lie product. Using equations (A18), (A17) and (A16), we obtain:

$$\frac{\partial f_m^{\epsilon_k \delta_\ell}}{\partial \phi_n} = \frac{1}{2}\mathbb{R}\left[\mathrm{Tr}\left(U_m^{\epsilon_k \delta_\ell}\right)^* \mathrm{Tr}\left(\underbrace{P_{mN}^{\epsilon_k \delta_\ell} \cdots P_{n+1}^{\epsilon_k \delta_\ell}}_{\Lambda_{m,n+1}^{\epsilon_k \delta_\ell}} \left[\frac{-i\sigma_z}{2}, P_n^{\epsilon_k \delta_\ell}\right] \underbrace{P_{n-1}^{\epsilon_k \delta_\ell} \cdots P_1^{\epsilon_k \delta_\ell}}_{\hat{U}_{n-1}^{\epsilon_k \delta_\ell}}\right)\right]. \tag{A19}$$

Here, $\hat{U}_{n-1}^{\epsilon_k\delta_\ell}$ represents the evolution operator obtained after applying the first $n-1$ π-pulses. The operator $\Lambda_{m,n+1}^{\epsilon_k\delta_\ell}$ is often referred to as *costate* or *adjoint state*, and is obtained using a backward propagation.

3. Global optimization via genetic algorithm

While the GRAPE algorithm explained in the previous section can efficiently compute the gradient of the total fidelity F with respect to the pulse phases ϕ_n, and hence allows for implementing a gradient-based local optimization, it turns out that the cost function contains many local minima that are far from being global minima. Hence, the final result strongly depends on the initial pulse. A possible approach is to use many random initial pulses, optimize them all using GRAPE, and simply choose the best final pulse. Using the problem structure stemming from tracking fidelity, we can improve on this naive approach.

The idea is to use a genetic algorithm, in conjunction with GRAPE, to optimize a *population* of pulses over several *generations*. Note that while genetic algorithms have been used previously in quantum control and dynamical decoupling [65, 66], combining them with local optimization via GRAPE significantly improves the results. The general structure of the algorithm is as follows:

Genetic algorithm.

(1) Generate a population of J random initial pulses $\tilde{\phi}_n^{(0,j)}$ with $j = 1, \ldots, J$.

(2) Locally optimize each pulse using GRAPE to obtain the pulses $\phi_n^{(0,j)}$. This is the 0-th generation.

(3) For $i = 1, \ldots, I$:

  (a) Apply a genetic update to the population $\phi_n^{(i-1,j)}$ to obtain $\tilde{\phi}_n^{(i,j)}$.

  (b) Locally optimize all pulses using GRAPE to obtain $\phi_n^{(i,j)}$.

Now, let us explain the genetic update. This step is based on the symmetry properties explained in Sec. III.

1. *Elitism:* Keep the best performing pulses. This guarantees that the best fidelity of each generation does not get worse.

2. *Mutation:* Some pulses are randomly modified in the following ways:

  (a) Some of the blocks are permuted randomly.

  (b) Some blocks are modified by flipping, reversing, or cyclically permuting the phases (note that we don't include a phase shift since this is a continuous parameter which is taken care of by the GRAPE algorithm)

3. *Recombination:* We randomly choose pairs of pulses and swap some of their blocks.

The key property of the genetic update is that a good pulse remains good under mutation, and two good pulses remain good under recombination. In this way, the algorithm does not search the entire space of all possible pulses, as happens with random pulses, but rather focuses on a subset of pulses that all perform well. As a result, the cost function improves significantly over the course of several generations, and the final optimized pulses obtain a nice structure as illustrated in Figure 6.

4. Highly parallel implementation

The optimization algorithm described in the previous two sections lends itself to parallelization. The GRAPE's gradient computation is performed in two steps. A forward and a backward propagation. The same operation must be performed for each value of the amplitude error and detuning $(\epsilon_k, \delta_\ell)$. Using a GPU, all of these can be performed in parallel, and the final contributions to the gradient are simply added up. Moreover, GRAPE can be applied to many pulses within a population in parallel, as the computations are independent.

Appendix B: Experimental Robustness Characterization of DD Sequences

This appendix describes the experimental implementation of the dynamical decoupling (DD) sequences studied in this work, as well as the robustness characterization measurements performed on a transmon qubit processor fabricated at the Walther-Meißner-Institut (WMI). The fidelity of each sequence was assessed via quantum process tomography (QPT) [55].

The DD sequences were implemented on a superconducting transmon qubit device [67]. In particular, we used one qubit of a chip containing 17 qubits and 24 couplers in a square-grid architecture, with qubit frequency $\omega_q/2\pi = 5497.798(2)\,\mathrm{MHz}$. The coherence parameters were $T_1 = 19.9(28)\,\mu\mathrm{s}$, $T_2^{\mathrm{Echo}} = 37(8)\,\mu\mathrm{s}$, and $T_2^{\mathrm{R}} = 29(6)\,\mu\mathrm{s}$. The experimental setup and wiring follow the description in [68].

The qubit drive was generated using an SHFSG from Zurich Instruments [69]. Each π pulse was implemented using a 4σ Gaussian envelope with a duration of 128 ns, modulated at the driven qubit frequency. The π-pulse amplitude and drive frequency were calibrated using error amplification sequences.

The sequences were implemented via calibrated π-X rotations, with appropriate pre- and post-oscillator phase increments. This realizes the desired equatorial-axis π rotation via $X_\theta \equiv R_z(\theta)\, X\, R_z(-\theta)$, where $X \equiv R_x(\pi)$. Deviations from an ideal X rotation must be accounted for in the DD sequence. Consequently, the simplified rearrangement $R_z(\theta)\, X\, R_z(-\theta) \propto X\, R_z(-2\theta)$, which is valid only for a perfect X gate up to a global phase, is in general not applicable in the presence of pulse imperfections.

On the FPGA-based control instrument, only a single waveform is stored, and the oscillator phase increments are applied by iterating consecutive command table entries.

To reconstruct the experimental process implemented on the device, we performed single-qubit quantum process tomography. We prepared the four input states $|0\rangle$, $|1\rangle$, $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$, and $|-i\rangle = (|0\rangle - i|1\rangle)/\sqrt{2}$. For each input state, we measured expectation values in the x, y, and z bases by applying post-rotation gates. Using these measurements, we reconstructed the quantum channel matrix following the standard protocol described in Ref. [55] and computed the process fidelity [70]. No additional post-processing was performed after QPT. Therefore, the differences in fidelity color intensity between experiment and simulation in Fig. 3 directly reflect the underlying process fidelities and are not influenced by any post-processing corrections.

To measure sequence detuning, the I-Q envelope of the Gaussian pulses was modulated at the desired offset frequencies, and corresponding oscillator phase shifts were applied. Static amplitude shifts were realized by scaling the pulse envelope. The pre- and post-rotations used for process tomography were performed using the calibrated gates.

Appendix C: Simulation of transmon qubits

1. Evolution operator

In the methods described in Appendix A, Eq. (A1) represents the evolution of a two-level system driven by a constant π-pulse subject to detuning and amplitude errors. In practice, however, all DD sequences in this work are implemented on transmon-based quantum processors, where the π-pulses have a finite duration and a smooth envelope, and where higher-energy levels of the transmon can be populated. Consequently, while Eq. (A1) provides a reasonable approximation of the dynamics, it does not fully capture effects arising from the pulse shape or from the presence of higher excited states.

To accurately reproduce the experimental results, we simulate the qubit dynamics under the pulse shape used in the experiment by numerically solving the time-dependent Schrödinger equation,

$$\dot{\tilde{P}}(t) = -iH(t)\tilde{P}(t), \tag{C1}$$

where $\tilde{P}$ is the evolution operator and $H(t)$ is the Hamiltonian in the rotating frame under the rotating-wave approximation:

$$H(t) = \tfrac{1+\epsilon}{2}\big(\Omega(t)a^\dagger + \Omega^*(t)a\big) + \delta\, a^\dagger a + \tfrac{\alpha}{2}\, a^{\dagger 2}a^2. \tag{C2}$$

Here, $\Omega(t) = \Omega_I(t) + i\,\Omega_Q(t)$ is the complex control pulse, with $\Omega_I(t)$ and $\Omega_Q(t)$ denoting the in-phase and quadrature components, respectively. The parameter α is the transmon anharmonicity, with $\alpha/(2\pi) = -173$ MHz, and a ($a^\dagger$) are the annihilation (creation) operators. The pulse envelope is the one used in the experiment (see Appendix B), namely a truncated Gaussian pulse without quadrature component:

$$\begin{cases} \Omega_I(t) = A\left[e^{-(t-T_\pi/2)^2/(2\sigma^2)} - e^{-T_\pi^2/(8\sigma^2)}\right], \\ \Omega_Q(t) = 0, \end{cases} \tag{C3}$$

for $0 \le t \le T_\pi$, where $T_\pi = 128$ ns is the pulse duration and $\sigma = T_\pi/4$. The constant offset ensures that the envelope vanishes at $t = 0$ and $t = T_\pi$. The amplitude A

is set by the condition $\int_0^{T_\pi} \Omega_I(t)\,\mathrm{d}t = \pi$, such that the pulse implements a π rotation when $\delta = \epsilon = 0$. This gives

$$A/(2\pi) = 8.44 \text{ MHz}. \tag{C4}$$

In our simulations, the pulse is assumed to be piecewise constant with time steps of 0.5 ns, and we truncate the operators a and $a^\dagger$ to the first three energy levels. This yields a 3×3 propagator, $\tilde{P}(\delta, \epsilon, T_\pi)$, representing the numerical solution of Eq. (C1) under the pulse of Eq. (C3). The propagator corresponding to a π-rotation with phase ϕ_n is then given by

$$\tilde{P}_n(\delta, \epsilon) = e^{-i\phi_n a^\dagger a}\, \tilde{P}(\delta, \epsilon, T_\pi)\, e^{i\phi_n a^\dagger a}. \tag{C5}$$

Thus, after M cycles of N pulses, the evolution operator of the system is a 3×3 matrix given by:

$$\tilde{U}_M(\delta, \epsilon) = \tilde{P}_{M\cdot N}(\delta, \epsilon)\tilde{P}_{M\cdot N-1}(\delta, \epsilon) \cdots \tilde{P}_1(\delta, \epsilon). \tag{C6}$$

With this operator, the fidelity at (δ, ϵ) is computed in the computational subspace as:

$$f(\delta, \epsilon) = \frac{1}{4}\Big| \langle 0|\, \tilde{U}_M(\delta, \epsilon)\, |0\rangle + \langle 1|\, \tilde{U}_M(\delta, \epsilon)\, |1\rangle \Big|^2. \tag{C7}$$

Population leaked to $|2\rangle$ reduces this fidelity. As shown in Fig. 3, this technique provides an excellent match with the experimental data.

Appendix D: Comparison with MLEV64 and XY64

In this section, we compare U-TRACK4 to two existing sequences from the literature built from short blocks applied with phase shifts between blocks: MLEV64 [8] and XY64 [9, 54]. All sequences are truncated to 40 π-pulses (10 blocks of 4 π-pulses) to match the same idle time used throughout the paper. Simulations are performed on a transmon-type Hamiltonian, where each π-pulse is implemented as a Gaussian pulse, as described in Appendix C.

As for the robustness profile across a range of parameter deviations shown in Fig. 3, we plot a simulated robustness profile for MLEV64, XY64, and U-TRACK4 sequences in Fig. 7, showing the fidelity after each block of 4 π-pulses. Overall, all three sequences achieve comparable fidelity profiles. However, MLEV64 shows a noticeable degradation at $t = 20T_\pi$ and $t = 24T_\pi$, which directly impacts the results of Fig. 8, as discussed below. XY64 does not suffer from this issue and maintains performance comparable to U-TRACK4 across all 10 blocks.

To provide a better overview, we reproduce Fig. 1, this time extending to 16 blocks (i.e., 64 pulses), for the three sequences mentioned above. The results are shown in Fig. 8, where each pixel of coordinate (n,m) shows the average fidelity over a range of robustness:

$$\bar{F}_{nm} = \eta \int_{\delta_{\min}}^{\delta_{\max}} \int_{\epsilon_{\min}}^{\epsilon_{\max}} \frac{1}{4}\Big| \langle 0|\, U_{\epsilon\delta}(t_n, t_m)\, |0\rangle + \langle 1|\, U_{\epsilon\delta}(t_n, t_m)\, |1\rangle \Big|^2 d\epsilon\, d\delta, \tag{D1}$$

where, as in Eq. (C7), the fidelity is evaluated in the computational subspace, $\eta = 1/[(\epsilon_{\max} - \epsilon_{\min})(\delta_{\max} - \delta_{\min})]$, and $U_{\epsilon\delta}(t_n, t_m)$ represents the evolution operator between time points t_n and t_m. In other words, it measures how well the identity gate performs on the pulse segment between t_n and t_m. We take the average over $[\delta_{\min}/(2\pi), \delta_{\max}/(2\pi)] \in [-1.5625, 1.5625]$ MHz (optimized range) and $\epsilon \in [-0.4, 0.4]$.

The MLEV64 profile exhibits prominent blue lines, indicating a sharp drop in fidelity at specific time points — notably at t_5 ($t = 20\,T_\pi$), t_6 ($t = 24\,T_\pi$), t_{13}, and t_{14}. These correspond to pulse segments that fail to implement the identity gate robustly; this can also be observed directly in the robustness plots at $t = 20\,T_\pi$ and $t = 24\,T_\pi$ of Fig. 7. Consequently, the average fidelity over the chosen parameter range is significantly degraded at these points.

Appendix E: Experimental comparison of DD sequences

To experimentally evaluate the performance of the U-TRACK4 and XY4 dynamical decoupling (DD) sequences, we conducted a targeted experiment designed to quantify their ability to preserve a specific quantum state. In analogy to the procedure used for the results presented in Fig. 3, we performed full quantum process tomography following the methodology detailed in Appendix B. In contrast to the experiments in Sec. II B, where amplitude deviations (ϵ) and static detuning (δ) were intentionally introduced to test robustness, here no such modifications were applied, and the sequences were tested under the system's natural conditions.

All experiments were performed on the 20-qubit superconducting device IQM Garnet [60], in which both computational qubits and tunable couplers are implemented as flux-tunable transmon qubits. For this study, we selected five qubits on the device, as indicated in Fig. 9. Each single-qubit gate has a duration of 24 ns. Full quantum process tomography was performed simultaneously on all five qubits. Specifically, we prepared the four input states $|0\rangle, |1\rangle, |+\rangle, |-i\rangle$, applied the DD sequences with varying numbers of cycles, and measured expectation values in the x, y, and z bases. In the "No-DD" case, no gates were applied, and the qubits evolved freely for the same duration as the corresponding DD pulse block. The experiment was repeated across five different days (spanning approximately eight calibration cycles).

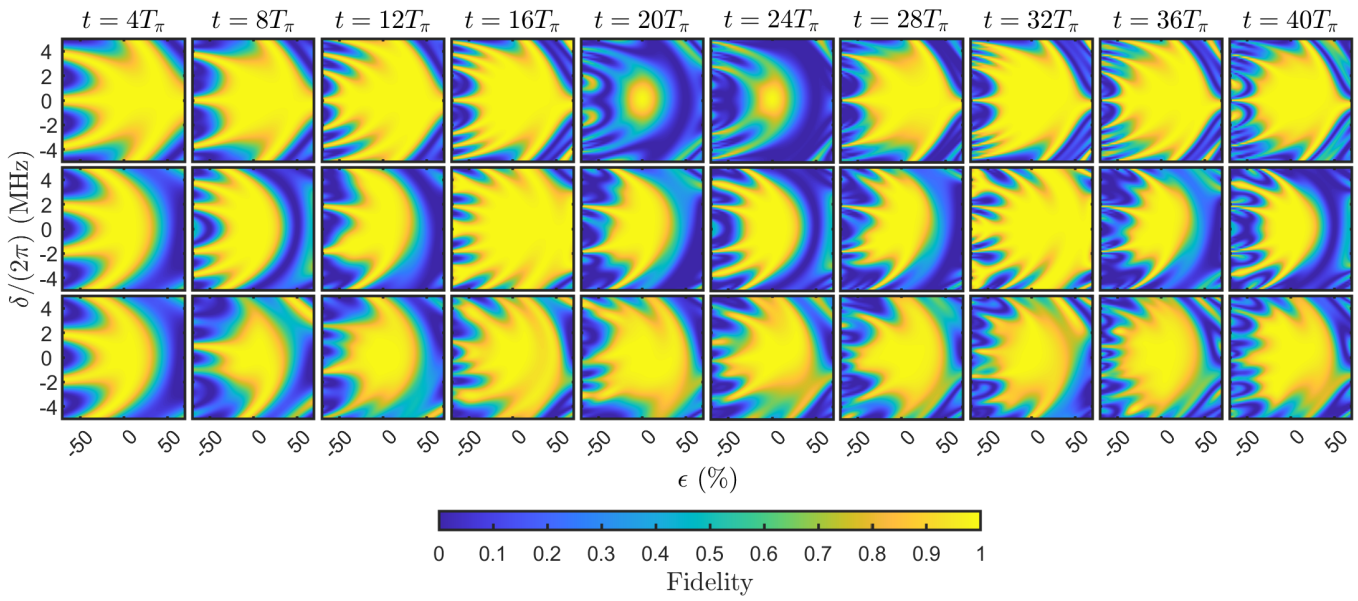


FIG. 7. Robustness profile of MLEV64 (top row), XY64 (middle row), and U-TRACK4 (bottom row), as functions of pulse-amplitude error ϵ and frequency detuning δ. Each column corresponds to the fidelity after a given number of blocks of 4 π-pulses, as indicated at the top. Each π-pulse has a duration of $T_\pi = 128$ ns. All sequences are truncated to 40 π-pulses (10 blocks of 4 π-pulses).

Fig. 10 reports the mean and standard deviation of the process fidelity $\mathcal{F}$ as a function of time, averaged over all qubits and days.

The top panel of Fig. 10 shows the results obtained when the DD pulses were applied without any inter-pulse delays. In this regime, the XY4 sequence performed worse than applying no DD at all. This behavior is likely caused by pulse distortions that lead to partial overlap between successive pulses when the pulse intervals are too short, resulting in shifts in the effective rotation axis [48, 53]. In contrast, the U-TRACK4 sequence demonstrated robustness to such distortions and exhibited significantly improved performance. We also observed similar effects on IBM quantum devices. A detailed investigation of these effects lies beyond the scope of this work.

We repeated the experiment with an added inter-pulse delay of $\tau = 40$ ns between each DD pulse (P), with the resulting sequence taking the form $P_1 - \tau - P_2 - \tau - P_3 - \tau - P_4 - \tau$. The corresponding results are shown in the bottom panel of Fig. 10. In this case, the performance of U-TRACK4 and XY4 was comparable, indicating that both sequences provide similar protection of quantum states when pulse overlap effects are mitigated. These results constitute preliminary observations; a more detailed experimental analysis will be presented in future work.

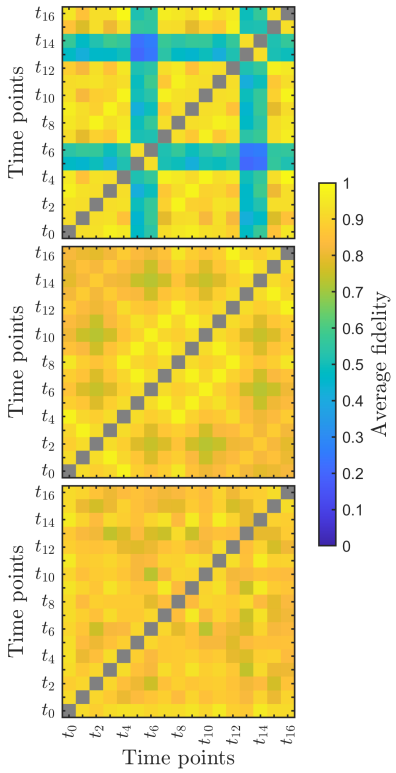


FIG. 8. Average fidelity of the identity gate as a function of pulse-segment boundaries for MLEV64 (top), XY64 (middle), and U-TRACK4 (bottom). All sequences here are shown for 64 π-pulses (16 blocks of 4 π-pulses). The time coordinate t_n denotes the time after n blocks of four π-pulses. Each pixel at coordinates (t_m, t_n) displays the average fidelity $\bar{F}_{mn}$ (Eq. D1), which quantifies how well the pulse segment (t_m, t_n) implements the identity gate despite frequency-offset and amplitude errors. Trivial diagonal entries corresponding to $U(t_n, t_n) = \mathbb{I}$ are grayed out.

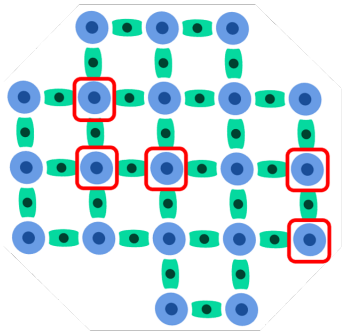


FIG. 9. Topology of the 20-qubit IQM Garnet device. The five qubits enclosed by the red box indicate the subset used in the experiments.

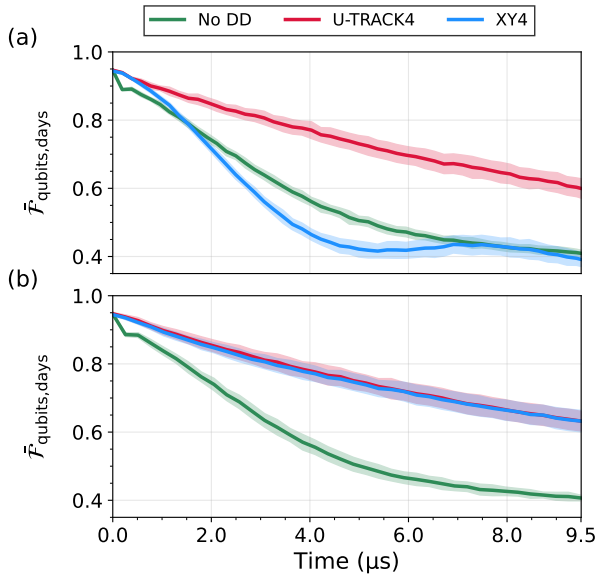


FIG. 10. Performance comparison of U-TRACK4 and XY4 dynamical decoupling sequences. (a) Process fidelity $\mathcal{F}$ as a function of total sequence duration without inter-pulse delays. (b) Process fidelity $\mathcal{F}$ as a function of total sequence duration with inter-pulse delays inserted between π pulses. In both panels, solid lines show the average fidelity $\bar{\mathcal{F}}$ over qubits and measurement days, and shaded regions indicate the corresponding standard deviation across days.