Japanese Journal of Applied Physics **64**, 02SP22 (2025) **REGULAR PAPER**
https://doi.org/10.35848/1347-4065/adafb3

Ray-based automatic tuning of a single quantum dot in a GaAs/AlGaAs quadruple dot array

Ehsan Alizadeh Kashtiban[1,2*], Takafumi Fujita[2,3,4,5,6], and Akira Oiwa[2,4,5,6]

[1]Department of Physics, Graduate School of Science, Osaka University, Toyonaka, Osaka, 560-0043, Japan
[2]SANKEN, Osaka University, Ibaraki, Osaka 567-0047, Japan
[3]Artificial Intelligence Research Center, The Institute of Scientific and Industrial Research, Osaka University, 8-1 Mihogaoka, Ibaraki, Osaka 567-0047, Japan
[4]Center for Quantum Information and Quantum Biology, Osaka University, 1-2 Machikaneyama, Tokyonaka, Osaka 560-0043, Japan
[5]Center for Spintronics Research Network, Graduate School of Engineering Science, Osaka University, 1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan
[6]Spintronics Research Network Division, Institute for Open Transdisciplinary Research (OTRI), Osaka University, Osaka 565-0871, Japan

*E-mail: alizadeh21@sanken.osaka-u.ac.jp

Received October 7, 2024; revised January 26, 2025; accepted January 29, 2025; published online February 13, 2025

Semiconductor quantum dot arrays, a key platform for spin-based quantum computing, face scalability challenges due to the time-consuming manual tuning of potential landscapes. This paper proposes that combining minimal one-dimensional voltage sweeps with machine learning offers a practical method for automating the tuning of four separate single quantum dots, inspired by the Ray-based Classification framework. Our method successfully tunes single quantum dots in a GaAs quadruple quantum dot device, while reducing the complexity of data acquisition and simplifying the tuning process. © 2025 The Author(s). Published on behalf of The Japan Society of Applied Physics by IOP Publishing Ltd

1. Introduction

Arrays of gate-defined quantum dots (QDs) are emerging as a promising candidate for storing spin qubits and serve as a platform for quantum computing applications.[1–3] Their promising potential arises from the ease of controlling key parameters,[4–7] the ability to rapidly measure spin and charge states,[8] and the long decoherence times they offer.[9–11] Previous advancements, including the demonstration of two-qubit gates and quantum algorithms,[12–14] have further strengthened their potential as a building block for solid-state quantum computers.

However, one of the primary challenges in using gate-defined QDs for large-scale quantum computing is the complex and time-consuming process of tuning these dots for QD array formation.[15–18] In this paper, tuning refers to the adjustment of the voltages applied to the electrostatic gates, which must be precisely controlled to trap individual electrons within the QDs. This process is crucial because each qubit must be in a well-defined quantum state to perform reliable quantum operations. Typically, the manual tuning process involves scanning the parameter space of gate voltages and observing the behavior of the QDs to identify the desired operating regime. However, this process is highly time-intensive, especially as the number of qubits in a system increases.

Several automated methods have been developed to tackle the challenges of tuning in quantum dot (QD) devices, aiming to simplify various steps in the tuning process. The choice of automation often depends on the device design, as each tuning step requires specific techniques. Some methods focus on the initial stages of tuning, transitioning the device from its starting configuration to a voltage regime where QDs can form.[16] Others concentrate on more complex tasks, such as configuring quantum dots—for instance, transitioning from two single quantum dots to coupled double quantum dots.[19] These methods aim to control the number of electrons within each QD or fine-tune the couplings in multi-QD systems.[20–23] They utilize a range of tools, including convolutional neural networks (CNNs),[24] deep generative modeling,[25] and classical feature extraction methods like the Hough transformation.[26]

Among the automated tuning methods developed for optimizing quantum dot devices, the Ray-Based Classification (RBC) framework offers a novel approach to reduce the measurement time and complexity involved in quantum dot tuning.[27] Traditional tuning methods often rely on exhaustive two-dimensional (2D) voltage scans, where gate voltages are varied systematically across a grid. This process is time-consuming and inefficient because the number of measurements increases significantly with the resolution of the scan. While 2D scans provide comprehensive information about the system, the RBC framework minimizes the number of measurements required by performing one-dimensional (1D) voltage sweeps along various directions, known as "rays." These rays exploit the freedom of angle in the multidimensional voltage space, allowing for efficient exploration with significantly fewer measurements. By strategically selecting the angles of these rays, the RBC framework balances the need for valuable system information with the practical constraints of time and resource efficiency.

By focusing on distinguishing between different charge states during quantum dot tuning, these rays provide qualitative information about the boundaries that separate various operational regimes of the quantum dots. This targeted approach enables a more efficient exploration of the parameter space and a faster identification of the desired charge configuration. In this way, the RBC framework complements existing methods by offering a more practical and time-effective means of achieving precise control over quantum dot charge states.

In this paper, operating within the RBC framework, we employ a Long Short-Term Memory (LSTM) neural network

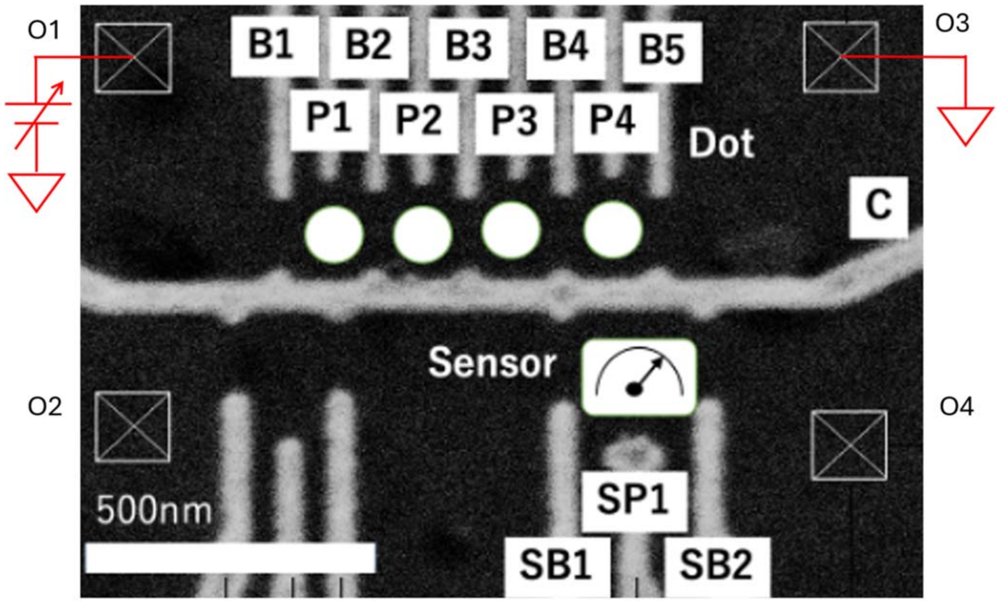

Fig. 1. A scanning electron microscope image of the GaAs/(Al,Ga)As sample. The circles indicate the positions of the four QDs that can be formed. The crossed boxes indicate ohmic contacts.

to create a closed-loop system for achieving single-dot formation in a GaAs/AlGaAs device.[28] LSTM networks are well-suited for handling sequential data and have proven effective in identifying patterns in time-series data.[29–31] By integrating the RBC framework's minimal one-dimensional sweeps with the LSTM neural network's pattern recognition capabilities, we develop a closed-loop system for automatically tuning independent single quantum dots in a quadruple dot device.

Our autotuning algorithm operates with minimal prior knowledge, specifically the gate design parameters and the measured depletion value of a central gate shared by all quantum dots. We detail the algorithm and validate it by successfully tuning each of four independent single quantum dots within a quadruple dot array. The algorithm autonomously identifies the appropriate gate voltages needed to tune each quantum dot into a separate single-dot regime, one at a time, as confirmed by experimental results.

2. Experimental methods

2.1. Experimental setup

Figure 1 shows a scanning electron micrograph of the device capable of forming up to four QDs.[32] The QD is electrostatically defined in a two-dimensional electron gas, situated 90 nm beneath the surface of a GaAs/AlGaAs heterostructure, using Ti/Au gate electrodes.

The operation of the QDs is controlled by several gates: the center gate (C), which separates the QD array from the sensor QD; the barrier gates (B1, B2, B3, B4, B5), which primarily adjust the tunnel rates between the dots and the reservoirs or between the dots themselves; and the plunger gates (P1, P2, P3, P4), which mainly regulate the chemical potential of each individual QD. The sensor dot gates (SB 1, SB 2, SP1) configure a sensitive charge sensor based on a conventional QD setup. All measurements were conducted at a temperature of 8 millikelvin.

The key transport feature indicating single-electron transport is the presence of Coulomb peaks, which manifest as peaks in the current flowing through the device as a function of a single plunger gate voltage. To reliably detect these Coulomb peaks and achieve single-electron transport, a classifier capable of differentiating between traces with and without these peaks is required. For this purpose, a Long Short-Term Memory (LSTM) neural network was trained using current traces obtained from the QD device, both with and without FFT preprocessing.

2.2. Experimental data acquisition

The training dataset consisted of 500 current traces collected from four distinct quantum dots, each sampled in 1 mV steps from −200 mV to 0 mV, resulting in a consistent length of 200 data points. A 1 mV step was chosen to provide sufficient resolution of Coulomb features without making measurements prohibitively long. Of these traces, 250 contained Coulomb peaks, serving as positive examples (Coulomb Peak Traces), and 250 lacked peaks, serving as negative examples (Non-Peak Traces). The fixed sweep length was selected to meet the LSTM's requirement for uniform input lengths, ensuring seamless integration with the neural network architecture. Out of these, 400 traces were used to train the LSTM model, while the remaining 100 traces were allocated for accuracy evaluation during the test phase. While 200 points effectively captured the necessary features for our GaAs/AlGaAs device, other quantum dot devices may necessitate shorter or longer sweeps. Figure 2 illustrates the data collection process, highlighting the distinction between Coulomb peak and non-peak traces.

During preprocessing, the current data were subjected to two distinct processing techniques. First, the raw current traces were used without any transformation. Second, a Fast Fourier Transform was applied to each current trace, and the resulting frequency-domain data were truncated to retain the first 200 frequency components. This truncation ensured that the raw and frequency-domain data had the same dimensionality, a necessary condition since both were used as input to the same LSTM model. To augment the training dataset and enhance the model's capacity to extract meaningful features

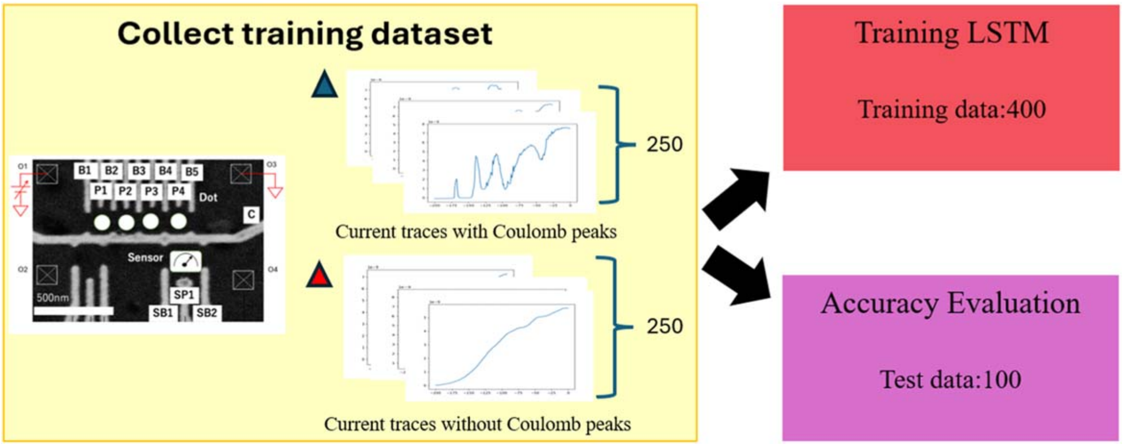

Fig. 2. Training dataset of 500 current traces (250 with Coulomb peaks and 250 without) used for LSTM model training (400 traces) and accuracy evaluation (100 traces).

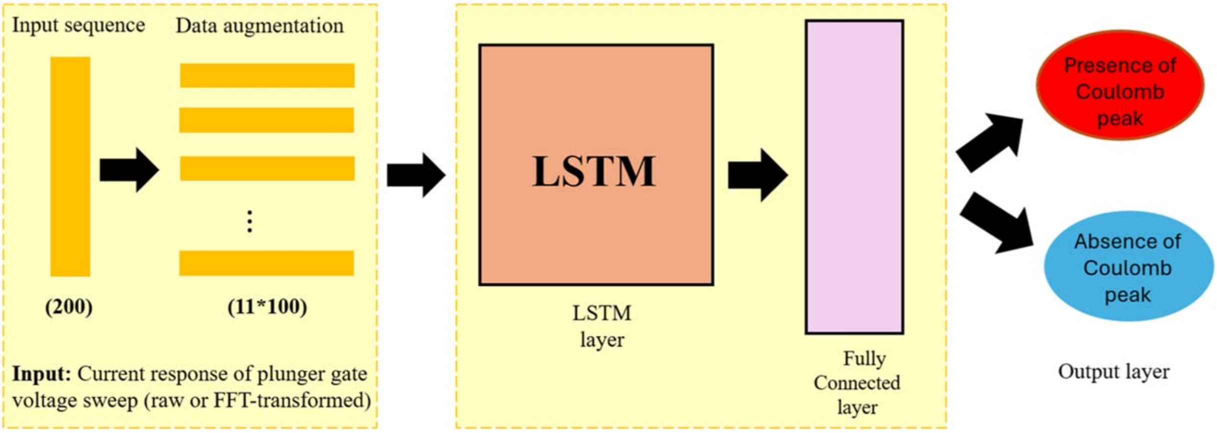

Fig. 3. Schematic of the LSTM model for Coulomb peak classification. The input, a current response from the plunger gate sweep, undergoes data augmentation and is processed by the LSTM network to classify the presence or absence of a Coulomb peak.

from the signal data, a data augmentation strategy was employed. Specifically, each current trace was segmented into overlapping windows, with each segment consisting of 100 data points and an overlap of 90 data points. This process transformed the original 200-point trace into 11 overlapping windows of 100 data points each. This data augmentation technique was applied to both the raw and FFT-transformed data, effectively increasing the size of the dataset available for model training.

After data augmentation, normalization was performed to scale all input values to a standard range before being passed to the LSTM neural network. This normalization process was conducted separately for the raw data and the FFT-transformed data. For each data type, the mean and standard deviation were calculated exclusively from the training dataset, ensuring that no information from the test dataset influenced the training process. These mean and standard deviation values were then applied to normalize the corresponding test datasets. By basing the normalization solely on the training data and treating raw and FFT-transformed data independently, we ensured that the model's performance evaluation remained unbiased.

The model architecture comprised a single LSTM layer with 128 hidden units, followed by a fully connected layer to classify traces as containing Coulomb peaks or not as depicted in Fig. 3. Training was conducted separately on the original and FFT-transformed datasets using the Adam optimizer, with a learning rate of 0.001 over 50 epochs. Cross-entropy loss was used to monitor both training and validation performance during each epoch to mitigate overfitting. The inclusion of FFT preprocessing resulted in significant improvements in classification performance, as summarized in Table I. Accuracy, which measures the proportion of correctly classified instances among all instances, increased from 64.0% to 94.0%. Precision, defined as the proportion of true positive predictions among all positive predictions made by the model, improved from 56.6% to 95.5%. Recall, the proportion of true positive predictions among all actual positive cases in the dataset, experienced a slight reduction from 93.5% to 91.3%. The F1 score, which is the harmonic mean of precision and recall and provides a balance between them, increased from 70.5% to 93.3%, demonstrating the efficacy of FFT in enhancing model performance.

It is important to note that precision and recall often have a trade-off relationship; improving one can sometimes lead to a decrease in the other. In our case, while precision significantly improved, indicating fewer false positives, recall

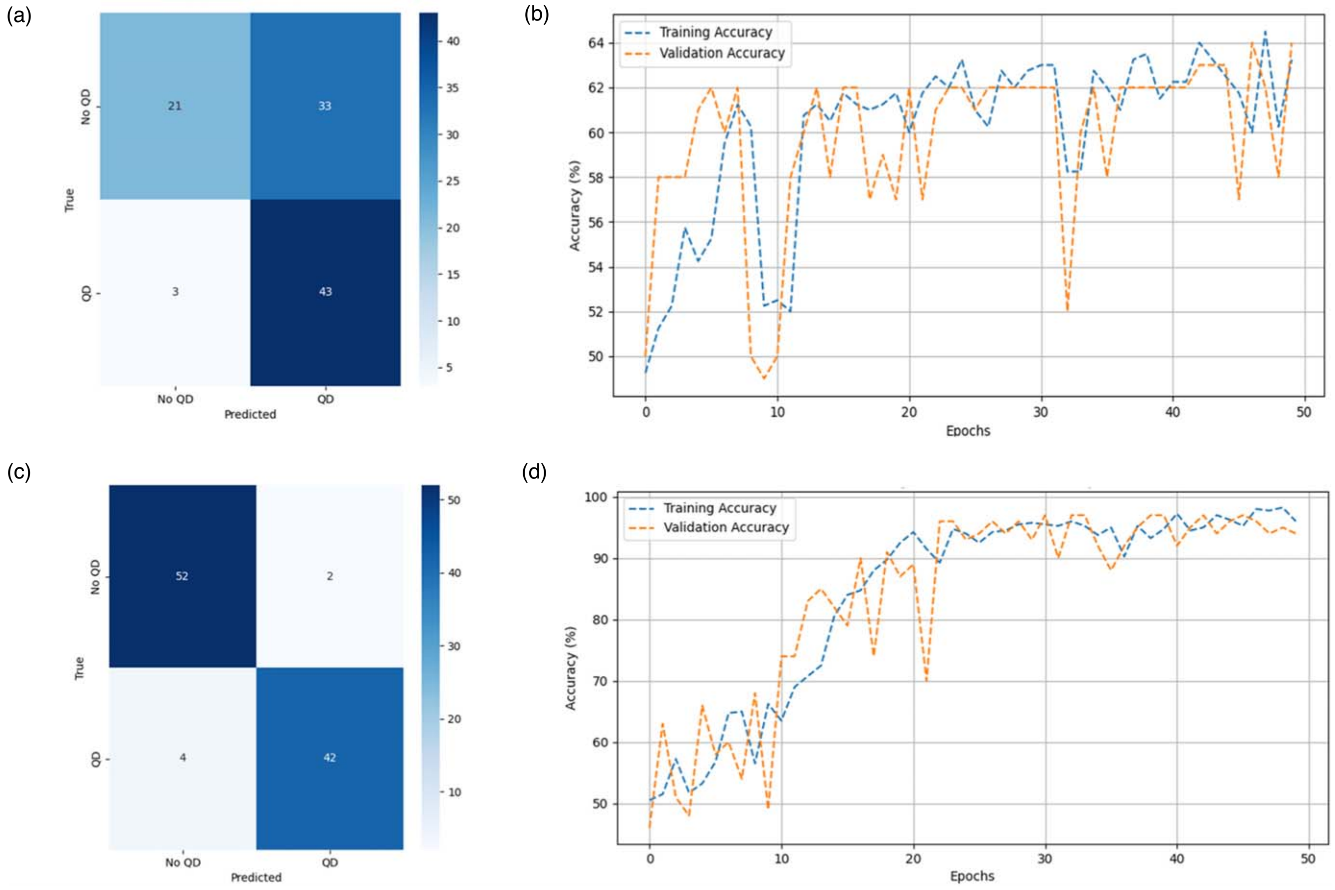

Fig. 4. (a) The confusion matrix for the model trained without FFT preprocessing shows lower classification accuracy, with high misclassification rates for both "No QD" and "QD" labels. (b) The corresponding training and validation accuracy curves over 50 epochs indicate slower convergence and fluctuations in performance when FFT preprocessing is not applied. (c) The confusion matrix for the model trained with FFT preprocessing shows higher classification accuracy, with significantly reduced misclassification rates for both "No QD" and "QD" labels. (d) The training and validation accuracy curves with FFT preprocessing exhibit faster convergence and more stable performance, ultimately reaching higher accuracy levels.

Table I. Performance metrics of the LSTM model without and with FFT preprocessing applied to current–voltage (I–V) traces. The application of FFT preprocessing significantly improves accuracy, precision, and F1 score.

Metric	Without FFT (%)	With FFT (%)
Accuracy	64.0	94.0
Precision	56.6	95.5
Recall	93.5	91.3
F1 Score	70.5	93.3

decreased slightly, suggesting a minor increase in false negatives. However, the substantial rise in the F1 score indicates that the FFT preprocessing contributed to a better overall balance between precision and recall, leading to more reliable classification results.

Figure 4(a) presents the confusion matrix for the model trained without FFT preprocessing, showing reduced classification accuracy. Significant misclassifications occur in both the "No QD" and "QD" categories, with 33 "No QD" instances misclassified as "QD" and 3 "QD" instances misclassified as "No QD." This reflects the model's difficulty in distinguishing between the two classes without the benefit of FFT preprocessing.

Figure 4(b) illustrates the training and validation accuracy curves over 50 epochs for the model without FFT preprocessing. The model reaches a plateau of around 62% accuracy; however, noticeable fluctuations indicate that it has not achieved robust convergence. While it does appear to stabilize around this level, the instability and variation between training and validation suggest that the model struggles to generalize well, showing signs of overfitting or underfitting.

Figure 4(c), on the other hand, displays the confusion matrix for the model trained with FFT preprocessing, demonstrating a marked improvement. Only 2 "No QD" instances and 4 "QD" instances are misclassified, highlighting the model's enhanced ability to differentiate between the two classes when FFT preprocessing is applied.

Figure 4(d) shows the training and validation accuracy curves with FFT preprocessing applied. These curves exhibit faster convergence, reaching higher accuracy levels and demonstrating more stability across epochs. This indicates better generalization and performance compared to the model without FFT, as the FFT preprocessing enables the model to achieve a more stable and higher level of accuracy.

Including the FFT preprocessing step increased the model's accuracy by transforming the current-versus-voltage (I-V) data into the frequency domain, thereby revealing underlying patterns and features that are not easily discernible in the original voltage domain. Although the data represents current as a function of voltage ($I(V)$) rather than time, applying FFT to this data allows us to analyze the

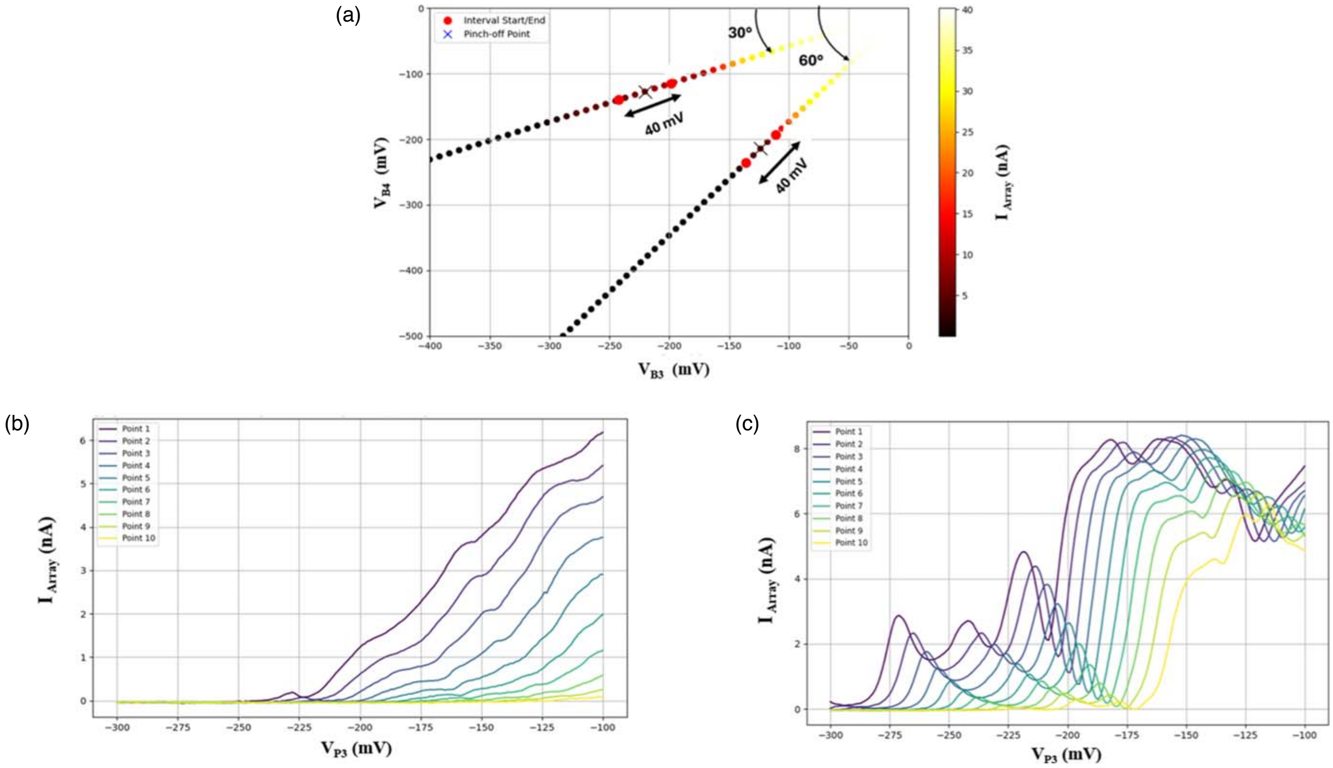

Fig. 5. (a) Rays at 30° and 60° in V_{B3}-V_{B4} space with pinch-off points identified by black crosses. A 40-mV interval around each pinch-off point (±20 mV) is selected for analysis. (b, c) Plunger gate voltage sweeps at 30° and 60° rays, respectively, showing the current responses at 10 points along each ray in the barrier voltage space.

signal's frequency components with respect to voltage changes.

Coulomb peaks in the *I-V* characteristics are associated with quantum effects that can introduce periodic variations in the current as the voltage is varied. These variations may be subtle and obscured by noise or other non-relevant fluctuations in the raw data. By performing FFT on the *I-V* data, we decompose the signal into its constituent frequency components, effectively highlighting these periodic features.

The first 200 frequency components extracted from the FFT provide a rich set of features that capture essential information about the presence of Coulomb peaks. These frequency-domain features can amplify the distinguishing characteristics between traces that contain Coulomb peaks and those that do not. In the frequency domain, the Coulomb peaks manifest as distinct frequency patterns or spikes, making them more detectable by the LSTM classification model. This benefits the LSTM neural network by providing inputs that emphasize frequency-based relationships within the data.

2.3. Automated tuning algorithm

Before the autotuning procedure begins, the user manually adjusts the center gate (C) to its depletion point, identified as the gate voltage at which current flow between the upper and lower channels is completely suppressed. To ensure effective isolation of the QD, a slightly more negative voltage than this depletion point is applied. While this step is currently performed manually, it could be automated in future implementations.

The autotuning process requires the user to specify the barrier and plunger gates that define the target QD (e.g., B2, P2, B3 for "QD2"). These gates will be actively adjusted by the algorithm, while all other gates are kept at zero voltage throughout the process. In addition, the user must define the voltage boundaries for the relevant barrier gates (e.g., $-400\,\mathrm{mV}$ on one gate and $-500\,\mathrm{mV}$ on another). This ensures that all measurements and voltage sweeps are conducted within a range where the operational "pinch-off" is guaranteed to occur.

With all relevant gate definitions and voltage boundaries established, the autotuning algorithm begins with an initialization stage where the user selects a small voltage bias (e.g., $\sim 200\,\mu\mathrm{V}$) to be applied to the Ohmic contact (O_1). Following this, the algorithm draws a ray at a randomly selected angle θ ($0° < \theta < 90°$) from the origin of the barrier voltage space. The intersection of this ray with the user-defined voltage boundaries is then calculated, and the algorithm uniformly samples a user-specified number of points (e.g., 50) along the resulting line segment.

After measuring the current at all sampled points, the data is smoothed using a moving average with a window size of five to reduce noise. Based on the smoothed current, the algorithm identifies the maximum measured current (I_{max}) and sets a threshold of $0.2 \times I_{max}$. The "pinch-off" voltage is defined as the first point along the ray where the smoothed current drops below this threshold. A user-defined voltage window around the pinch-off is then used to focus on this region in greater detail and is divided into a user-specified number of barrier-voltage setpoints (e.g., ten).

Figure 5(a) illustrates this process for two example rays, drawn at 30° and 60° in the V_{B3}-V_{B4} space. For each ray, 50 sampling points are used along each ray, and a ±20 mV window (total of 40 mV) around pinch-off is highlighted. The pinch-off points for each ray are marked with crosses, and the corresponding windows are indicated.

For each setpoint, the barrier voltages are held constant while the plunger gate is swept from −200 mV to 0 mV in 1 mV increments, producing 200 data points per sweep—precisely the input dimension required for the Long Short-Term Memory (LSTM) neural network.

For some rays, as shown in Fig. 5(b) for the 30° ray, clear Coulomb peaks are observed at each of the points along the ray, indicating the formation of a single dot. In contrast, Fig. 5(c) illustrates the current response along the 60° ray, where only weak Coulomb peaks are visible, highlighting the variability in the formation of the quantum dot across different rays. This variability underscores the need for automation in the tuning process. While the pinch-off point is consistently defined at 20% of the maximum current, the presence of Coulomb peaks during the plunger gate sweep is not guaranteed for all rays.

Each plunger sweep (current trace) is analyzed by a trained LSTM that determines whether Coulomb peaks are present. For each ray, the algorithm tests all equally spaced barrier-voltage setpoints within the user-defined window around the pinch-off voltage. If the LSTM classifies at least one setpoint as "QD," the quantum dot is considered successfully tuned into the single-dot regime, and the process ends.

If the LSTM identifies Coulomb peaks at any setpoint within the specified voltage range, the algorithm verifies the single-dot regime formation and applies the final gate voltages, concluding the successful tuning process. However, if every setpoint in the user-defined window is classified as "No QD," the ray is discarded, and the algorithm selects a new random angle in the barrier voltage space to measure the pinch-off and repeat the plunger gate sweeps. This iterative process continues until the LSTM network confirms that the single-dot regime has been achieved, at which point the final gate voltages are applied.

By systematically iterating over random ray angles, identifying the pinch-off voltage, defining a precise voltage window centered around the pinch-off for focused analysis, and conducting plunger gate sweeps within this targeted region, our method significantly reduces the need for exhaustive two-dimensional voltage scans. Furthermore, users retain control over key parameters, such as the number of points initially sampled along each ray to locate the pinch-off, the size of the voltage window around the pinch-off for detailed analysis, and the number of barrier-voltage setpoints sampled within this window—all of which can be defined based on prior experimental knowledge. This flexibility ensures adaptability to varying experimental conditions while leveraging domain expertise to optimize the process.

The LSTM's classification plays a critical role in the autotuning algorithm, enabling accurate identification of Coulomb peaks, which are indicative of single-dot regime formation. Its ability to reliably detect Coulomb peaks in current traces ensures robust tuning in the autotuning procedure.

The entire autotuning procedure is summarized in Fig. 6, which illustrates how the system automates ray selection, plunger gate sweeps, and state detection by the LSTM to achieve a single-dot regime.

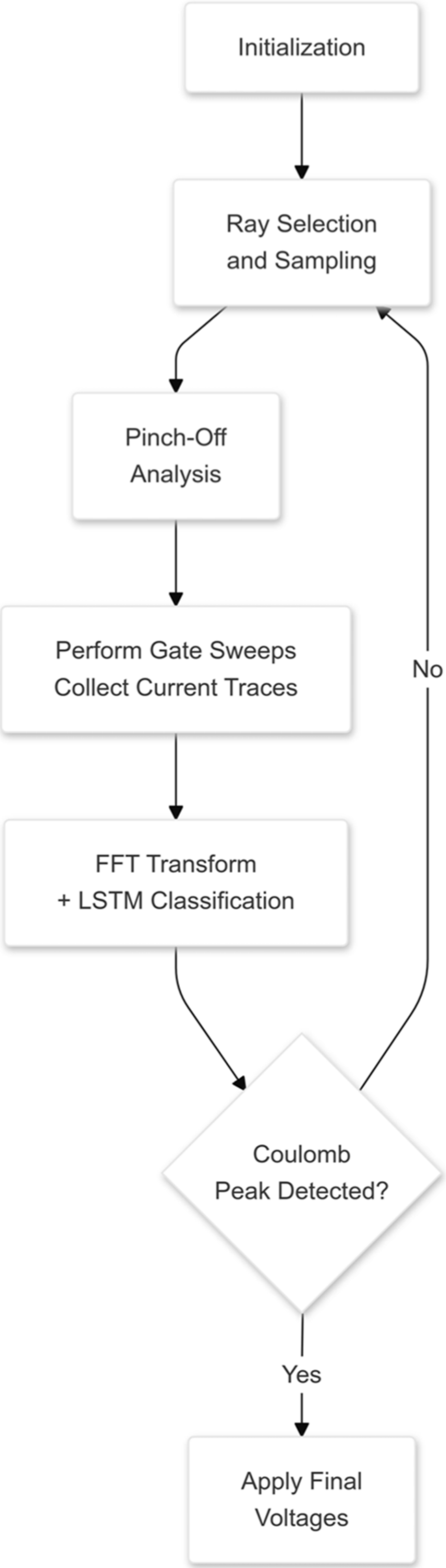
Initialization
Ray Selection and Sampling
Pinch-Off Analysis
Perform Gate Sweeps Collect Current Traces
FFT Transform + LSTM Classification
Coulomb Peak Detected?
No
Yes
Apply Final Voltages

Fig. 6. Flowchart of the automated quantum dot tuning algorithm. The process begins with initialization, followed by random ray selection and sampling in the barrier voltage space. Pinch-off voltages are determined from the measured current, and a voltage window around the pinch-off is defined for further analysis. Plunger gate sweeps are performed at setpoints within this window, and the resulting current traces are analyzed using FFT and an LSTM neural network. If Coulomb peaks are detected, confirming the single-dot regime, final gate voltages are applied. If no Coulomb peaks are detected, the process iterates with a new ray until successful tuning is achieved.

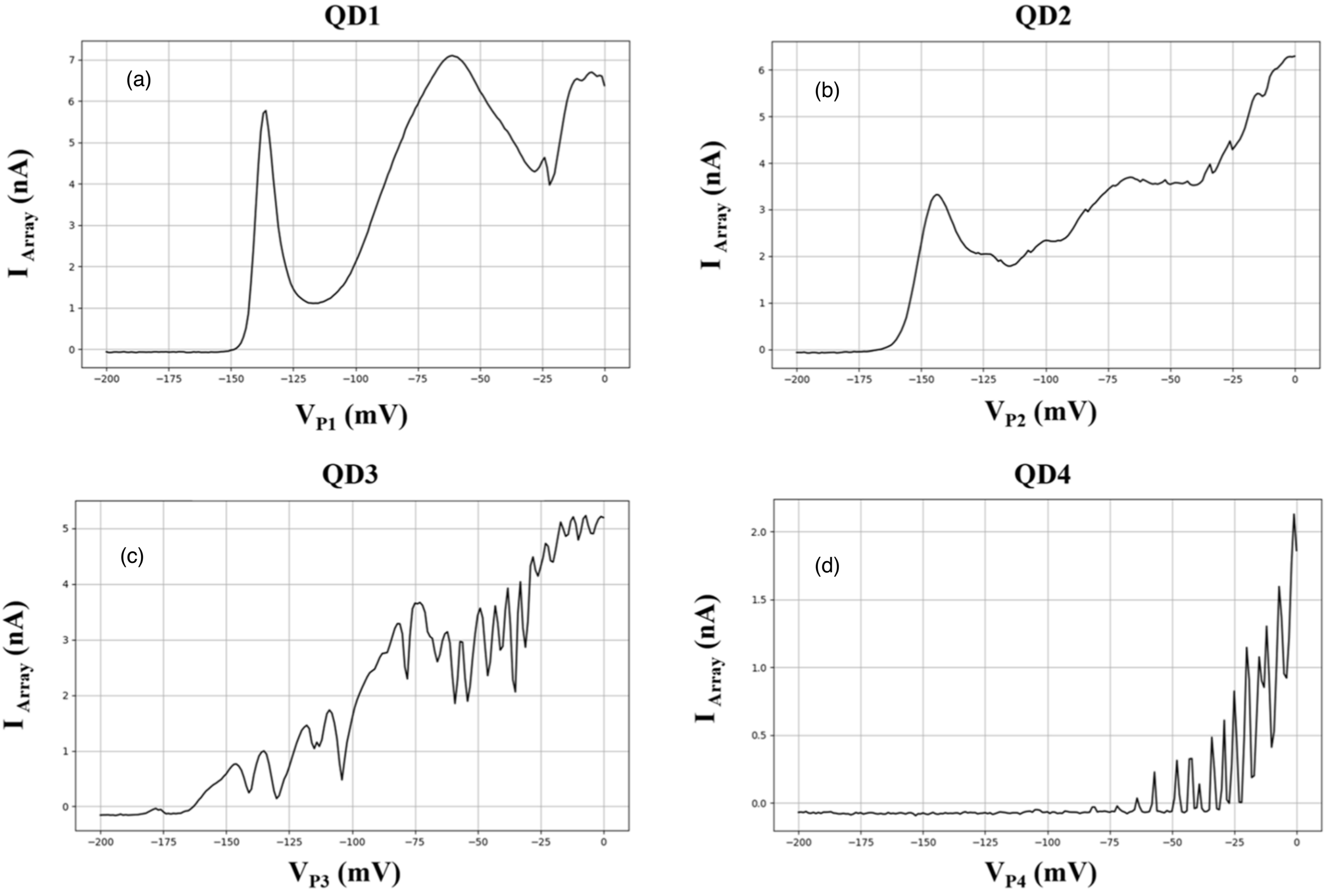

Fig. 7. Automated tuning results for GaAs quadruple quantum dot formation. The four plots (QD1, QD2, QD3, QD4) show current responses from gate voltage sweeps at angles of 23°, 83°, 49°, and 53°. Successful single quantum dot formation is confirmed by the presence of Coulomb peaks in the current traces.

3. Results and discussion

3.1. Automated tuning results

We successfully automated the tuning of a GaAs quadruple quantum dot into single-dot regimes using one-dimensional voltage sweeps combined with an LSTM neural network for classification. The LSTM model was trained on current traces exhibiting both the presence and absence of Coulomb peaks to effectively detect quantum dot formation.

During each trial, voltage sweeps were performed at three distinct points in the barrier voltage space. The LSTM model evaluated the probabilities of “No Quantum Dot” (No QD) and “Quantum Dot” (QD) formation at each point, denoted as $[P_{\mathrm{No\ QD}},P_{\mathrm{QD}}]$. The configuration with the highest probability for QD formation was selected for subsequent analysis.

As an example, we present the tuning process for Quantum Dot 1 (QD1). In the first trial with a ray angle of 54°, the LSTM model did not detect Coulomb peaks at any of the three points, yielding high probabilities for “No QD” formation: [0.998,0.002], [0.993,0.007], and [0.987,0.013] for each respective point. This indicates that the initial configurations were not conducive to quantum dot formation.

In the second trial, the ray angle was selected as 23°, which resulted in successful QD formation. The LSTM model’s evaluations for this trial were [0.999,0.001], [0.000,1.000], and [0.193,0.807]. The second point, with the highest QD probability of [0.000,1.000] was selected. This selection was confirmed by the presence of Coulomb peaks in the current trace, as depicted in Fig. 7(a).

This procedure was applied to each quantum dot in the array. By employing our method, we successfully formed four independent single quantum dots within the quadruple dot array. The presence of Coulomb peaks in each case confirmed successful single-dot formation.

Figure 7 presents the automated tuning results for the four single QD formations. The four plots (QD1, QD2, QD3, QD4) display current responses from gate voltage sweeps at angles of 23°, 83°, 49°, and 53°, respectively. The distinct Coulomb peaks observed in each trace confirm the successful formation of single quantum dots.

3.2. Discussion

In this work, we demonstrated a practical method for the automatic tuning of individual quantum dots in a GaAs/AlGaAs quadruple dot device by combining one-dimensional voltage sweeps with machine learning-based approaches. Our autotuning algorithm circumvents the conventional practice of measuring two-dimensional stability diagrams, which can be time-consuming and resource-intensive. By combining 1D sweeps with an LSTM-based classification of Coulomb peaks, we showed that it is possible to form single-dot states without exhaustive 2D parameter scans. While our method demonstrated the potential for reducing tuning overhead, several key steps in the preparation process remain manual. Initial gate adjustments, such as setting the center gate to its depletion point and conducting leakage tests, require user intervention. As the complexity of QD devices grows, manual tuning becomes increasingly infeasible, necessitating robust and scalable autotuning solutions. These steps are critical for ensuring the device's functionality could benefit significantly from automation.

Recent advancements provide frameworks to address these challenges. The Autonomous Bootstrapping Algorithm[33] offers a method for initializing a depletion mode QD device in preparation for subsequent phases of tuning. Similarly, BATIS[34] navigates high-dimensional gate voltage spaces, automating essential pre-tuning steps necessary such as leakage testing and gate characterization while offering platform agnostic adaptability for deployment across different QD devices. Together, these methods provide automatic "bootstrapping," a pre-tuning process that brings the device to an operational regime where the tuning stage can be initiated, serving as a baseline for the development of a fully autonomous QD device initialization and calibration process.

In our autotuning procedure, we relied solely on measuring current through the dot to identify Coulomb peaks and verify the single-dot regime. While effective in this work, this approach is not ideal for large-scale qubit tuning, where scalability and time efficiency are critical. For larger quantum dot arrays, RF-based tuning approaches may offer better functionality due to their high bandwidth and scalability through frequency multiplexing. An all-RF-based automated tuning method has already been proposed, eliminating the need for transport current measurements.[35]

In our algorithm, rays are drawn from the origin, and intersections with user-defined voltage boundaries are used to define line segments. Sampling is performed along these line segments, measuring current and defining pinch-off based on a fraction of the maximum current within the segment. While this approach effectively identifies pinch-off points, it requires sampling along the entire ray, which can be time-intensive and less efficient for large-scale applications.

To address this limitation, a hardware-triggered detection method based on reflectometry has been proposed, which identifies peaks along rays and terminates measurements as soon as a peak is detected. By eliminating the need to sample the entire ray, this method significantly reduces measurement time and enhances efficiency, making it highly suitable for scaling to larger quantum dot arrays.[36]

A key benefit of reducing the voltage space to a one-dimensional "ray" is that the number of required measurements grows only with the sampling resolution and the number of rays, rather than increasing quadratically as in traditional two-dimensional scans. However, relying on random ray angles has a drawback: not every angle successfully forms a single-dot state. In such cases, the autotuning algorithm abandons that ray and tests another, iterating until a single-dot regime is verified by the LSTM model. While this approach reduces measurement complexity, it does not fully leverage the feedback from rays that fail to produce Coulomb peaks, which could inform subsequent ray selections.

To address these limitations, we propose refining the ray-based strategy through adaptive sampling and feedback. Rays that fail to show strong Coulomb peaks still reveal valuable insights about the voltage space, which can guide future ray selections. A reinforcement learning (RL) framework could incorporate this feedback, enabling real-time adjustments to both the sweep angle and the sweep range to focus on more promising regions.

In conclusion, our method demonstrates that one-dimensional sweeps combined with machine learning provide a practical and efficient approach for automatic quantum dot tuning. By integrating recent advancements for automating initialization and gate characterization, hardware-triggered reflectometry to detect current peaks efficiently, and reinforcement learning to optimize sweep angles and ranges through adaptive sampling and real-time feedback, the tuning process could be further streamlined, enabling efficient, scalable tuning for large quantum dot arrays in practical quantum computing applications.

4. Conclusions

We have presented a procedure for automatically tuning four single QDs in a GaAs/AlGaAs quadruple QD device using one-dimensional voltage sweeps combined with an LSTM neural network. By building within the RBC framework and employing FFT preprocessing, our autotuning algorithm demonstrated high classification performance by accurately detecting the presence of Coulomb peaks in current traces. This enabled the successful formation of four independent single quantum dots within the quadruple dot device, as confirmed by experimental results. Our method requires minimal prior knowledge—specifically, the gate design and the measured pinch-off value of a central gate.

In conclusion, our work demonstrates that combining minimal one-dimensional voltage sweeps with machine learning provides a practical method for automating the tuning of quantum dots into single-dot regimes. While this approach simplifies the tuning process, further research is needed to fully explore its potential and to understand how it might be applied to larger quantum dot systems in the future.

Acknowledgments

We are deeply grateful to Dr. Yuta Matsumoto for his dedicated efforts in fabricating the device We also extend our gratitude to Professor Andreas D. Wieck and Arne Ludwig for providing the high-quality GaAs/AlGaAs wafer, which was instrumental in advancing our research.

We thank Mr. Hideaki Yuta, Mr. Tatsuo Tsuzuki, and Mr. Yuta Konishi for their support and contributions to this project. We also express our gratitude to Dr Rio Fukai for setting up the necessary infrastructure for the experiments.

This work was supported by JSPS KAKENHI (Grant No. JP23H05455) and JST Moonshot R&D (Grant No. JPMJMS226B).

1) D. Loss and D. P. DiVincenzo, Phys. Rev. A **57**, 120 (1998).
2) R. Hanson, L. P. Kouwenhoven, J. R. Petta, S. Tarucha, and L. M. K. Vandersypen, Rev. Mod. Phys. **79**, 1217 (2007).
3) F. A. Zwanenburg, A. S. Dzurak, A. Morello, M. Y. Simmons, L. C. L. Hollenberg, G. Klimeck, S. Rogge, S. N. Coppersmith, and M. A. Eriksson, Rev. Mod. Phys. **85**, 961 (2013).
4) J. R. Petta, A. C. Johnson, J. M. Taylor, E. A. Laird, A. Yacoby, M. D. Lukin, C. M. Marcus, M. P. Hanson, and A. C. Gossard, Science **309**, 2180 (2005).
5) F. H. L. Koppens, C. Buizert, K. J. Tielrooij, I. T. Vink, K. C. Nowack, T. Meunier, L. P. Kouwenhoven, and L. M. K. Vandersypen, Nature **442**, 766 (2006).
6) J. Medford, J. Beil, J. M. Taylor, E. I. Rashba, H. Lu, A. C. Gossard, and C. M. Marcus, Phys. Rev. Lett. **111**, 050501 (2013).

7) D. Kim, D. R. Ward, C. B. Simmons, D. E. Savage, M. G. Lagally, M. Friesen, S. N. Coppersmith, and M. A. Eriksson, npj Quantum Inf. **1**, 15004 (2015).

8) C. Barthel, D. J. Reilly, C. M. Marcus, M. P. Hanson, and A. C. Gossard, Phys. Rev. Lett. **103**, 160503 (2009).

9) M. Veldhorst et al., Nat. Nanotechnol. **9**, 981 (2014).

10) E. Kawakami, P. Scarlino, D. R. Ward, F. R. Braakman, D. E. Savage, M. G. Lagally, M. Friesen, S. N. Coppersmith, M. A. Eriksson, and L. M. K. Vandersypen, Nat. Nanotechnol. **9**, 666 (2014).

11) J. Yoneda et al., Nat. Nanotechnol. **13**, 102 (2018).

12) M. Veldhorst et al., Nature **526**, 410 (2015).

13) D. M. Zajac, A. J. Sigillito, M. Russ, F. Borjans, J. M. Taylor, G. Burkard, and J. R. Petta, Science **359**, 439 (2018).

14) T. F. Watson et al., Nature **555**, 633 (2018).

15) S. J. Darulová, N. Pauka, K. W. Wiebe, G. C. Chan, M. J. Gardener, M. C. Manfra, Cassidy, and M. Troyer, Phys. Rev. Appl. **13**, 054005 (2020).

16) H. Moon et al., Nat. Commun. **11**, 4161 (2020).

17) J. D. Teske, B. Thorgrimsson, J. P. Corrigan, M. P. Losert, C. DN, and L. M. K. Vandersypen, Appl. Phys. Lett. **114**, 133102 (2019).

18) T. Botzem, X. Fu, A. V. Kuhlmann, M. Veldhorst, H. Bluhm, and A. Ludwig, Phys. Rev. Appl. **10**, 054026 (2018).

19) J. P. Zwolak, T. McJunkin, S. S. Kalantre, J. P. Dodson, E. R. MacQuarrie, D. E. Savage, M. G. Lagally, S. N. Coppersmith, M. A. Eriksson, and J. M. Taylor, Phys. Rev. Appl. **13**, 034075 (2020).

20) T. Botzem et al., Phys. Rev. Appl. **10**, 054026 (2018).

21) C. J. van Diepen, P. T. Eendebak, B. T. Buijtendorp, U. Mukhopadhyay, T. Fujita, C. Reichl, W. Wegscheider, and L. M. K. Vandersypen, Appl. Phys. Lett. **113**, 033101 (2018).

22) J. D. Teske, S. S. Humpohl, R. Otten, P. Bethke, P. Cerfontaine, J. Dedden, A. Ludwig, A. D. Wieck, and J. H. Bluhm, Appl. Phys. Lett. **114**, 133102 (2019).

23) R. Durrer, B. Kratochwil, J. V. Koski, A. J. Landig, C. Reichl, W. Wegscheider, T. Ihn, and E. Greplova, Phys. Rev. Appl. **13**, 054019 (2020).

24) S. S. Kalantre, J. P. Zwolak, S. Ragole, X. Wu, N. M. Zimmerman, M. D. Stewart Jr, and J. M. Taylor, npj Quantum Inf. **5**, 6 (2019).

25) D. T. Lennon, H. Moon, L. C. Camenzind, L. Yu, D. M. Zumbühl, G. A. D. Briggs, M. A. Osborne, E. A. Laird, and N. Ares, npj Quantum Inf. **5**, 79 (2019).

26) A. R. Mills, M. M. Feldman, C. Monical, P. J. Lewis, K. W. Larson, A. M. Mounce, and J. R. Petta, Appl. Phys. Lett. **115**, 113501 (2019).

27) J. P. Zwolak, T. McJunkin, S. S. Kalantre, S. F. Neyens, E. R. MacQuarrie, M. A. Eriksson, and J. M. Taylor, PRX Quantum **2**, 020335 (2021).

28) E. Alizadeh Kashtiban, T. Fujita, and A. Oiwa, presented at SSDM 2024, Int. Conf. Solid State Devices and Materials, 2024, 10.7567/ssdm.2024.j-1-04.

29) S. Hochreiter and J. Schmidhuber, Neural Comput. **9**, 1735 (1997).

30) A. Graves, A. R. Mohamed, and G. Hinton, IEEE Int. Conf. Acoust. Speech Signal Process. **2013**, 6645 (2013).

31) F. A. Gers, J. Schmidhuber, and F. Cummins, Neural Comput. **12**, 2451 (2000).

32) Y. Matsumoto, T. Fujita, A. Ludwig, A. D. Wieck, K. Komatani, and A. Oiwa, npj Quantum Inf. **7**, 136 (2021).

33) A. Zubchenko, D. Middlebrooks, T. Rasmussen, L. Lausen, F. Kuemmeth, A. Chatterjee, and J. P. Zwolak, arXiv:2407.20061.

34) T. J. Kovach, D. Schug, M. A. Wolfe, E. R. MacQuarrie, P. J. Walsh, J. Benson, M. Friesen, M. A. Eriksson, and J. P. Zwolak, arXiv:2412.07676.

35) B. van Straaten, F. Fedele, F. Vigneau, J. Hickie, D. Jirovec, A. Ballabio, and N. Ares, arXiv:2211.04504.

36) A. Chatterjee, F. Ansaloni, T. Rasmussen, B. Brovang, F. Fedele, H. Bohuslavskyi, and F. Kuemmeth, Phys. Rev. Appl. **18**, 064040 (2022).