

A machine learning approach for automated fine-tuning of semiconductor spin qubits

Cite as: Appl. Phys. Lett. **114**, 133102 (2019); <https://doi.org/10.1063/1.5088412>

Submitted: 10 January 2019 . Accepted: 15 March 2019 . Published Online: 02 April 2019

Julian D. Teske , Simon Sebastian Humpohl , René Otten , Patrick Bethke, Pascal Cerfontaine ,
Jonas Dedden , Arne Ludwig , Andreas D. Wieck , and Hendrik Bluhm 



View Online



Export Citation



CrossMark

Applied Physics Reviews
Now accepting original research

2017 Journal
Impact Factor:
12.894

AIP
Publishing

A machine learning approach for automated fine-tuning of semiconductor spin qubits

Cite as: Appl. Phys. Lett. **114**, 133102 (2019); doi: [10.1063/1.5088412](https://doi.org/10.1063/1.5088412)

Submitted: 10 January 2019 · Accepted: 15 March 2019 ·

Published Online: 2 April 2019



Julian D. Teske,¹  Simon Sebastian Humpohl,¹  René Otten,¹  Patrick Bethke,¹ Pascal Cerfontaine,¹ 
Jonas Dedden,¹  Arne Ludwig,²  Andreas D. Wieck,²  and Hendrik Bluhm^{1,a)} 

AFFILIATIONS

¹JARA-FIT Institute for Quantum Information, Forschungszentrum Jülich GmbH and RWTH Aachen University, 52074 Aachen, Germany

²Lehrstuhl für Angewandte Festkörperphysik, Ruhr-Universität Bochum, 44780 Bochum, Germany

^{a)}Electronic mail: bluhm@physik.rwth-aachen.de

ABSTRACT

While spin qubits based on gate-defined quantum dots have demonstrated very favorable properties for quantum computing, one remaining hurdle is the need to tune each of them into a good operating regime by adjusting the voltages applied to electrostatic gates. The automation of these tuning procedures is a necessary requirement for the operation of a quantum processor based on gate-defined quantum dots, which is yet to be fully addressed. We present an algorithm for the automated fine-tuning of quantum dots and demonstrate its performance on a semiconductor singlet-triplet qubit in GaAs. The algorithm employs a Kalman filter based on Bayesian statistics to estimate the gradients of the target parameters as a function of gate voltages, thus learning the system response. The algorithm's design is focused on the reduction of the number of required measurements. We experimentally demonstrate the ability to change the operation regime of the qubit within 3–5 iterations, corresponding to 10–15 min of lab-time.

Published under license by AIP Publishing. <https://doi.org/10.1063/1.5088412>

The realization of qubits based on semiconductor quantum dots has reached a point where concrete architectures for large scale quantum processors are being considered.¹ Recent achievements representative of the state of the art include demonstrations of two-qubit gates,^{2–4} single qubit fidelities meeting the requirements for error correction,⁵ and first steps in operating arrays of quantum dots.^{6–8} Another important ingredient for scaleup is inter-qubit coupling over extended distances,^{8,9} e.g., via electron shuttling or cavities.¹⁰

A central starting point for the operation of qubits based on gate-defined quantum dots is the so-called *tuning* of the system, i.e., the procedure of identifying the voltages that need to be applied to the electrostatic gates to capture and to tunnel-couple individual electrons. Tuning by a human operator is very time consuming and will be impractical for multi-qubit systems with more than a handful of qubits. Efficient tuning is particularly pertinent for gate-defined quantum dots because of the large number of tunable dot parameters and gate voltages to control them, but is also relevant for other systems. For qubits based on quantum dots, tuning is a two-step procedure, which comprises both the formation of quantum dots and their depletion into the few electron regime—which we refer to as coarse-tuning—and, subsequently, the adjustment of parameters which define

the operation conditions of the qubit such as the tunnel coupling to leads and between dots. This procedure is referred to as fine-tuning of the qubit.

The initial coarse-tuning of gate-defined quantum dots relies mainly on the recognition of certain features in charge stability diagrams (CSDs), which reflects the dot occupancy as a function of gate voltages or transports through the dots.^{11,12} The most common practice is that a human operator interprets these measurements and decides based on experience and intuition of how to adjust gate voltages. Such manual tuning schemes have been extended to arrays of quantum dots^{7,8} using virtual gates that compensate for capacitive crosstalk between physical gates. Automated coarse tuning using image processing tools like the Gabor filter and template matching to identify quantum dot signatures has been demonstrated.¹³ Machine learning techniques like convolutional neural networks to quantify characteristics of CSDs by their similarity to simulated or measured reference CSDs were also explored.¹⁴

To guide the subsequent fine-tuning, one can use either transport measurements as well (e.g., supplementary material of Ref. 15) or a set of pulsed-gate experiments that extract the parameters of interest.^{16,17} Adjusting gate voltages based on this information is complicated by

the nonlinear dependence of the tunnel couplings on the gate voltages. Furthermore, gate-defined quantum dots exhibit a strongly coupled response, meaning that the voltage applied to each electrode has a considerable influence on several chemical potentials and tunnel couplings. “Virtual gates,” i.e., fixed linear combinations of gate voltages each of which predominantly affects a single qubit parameter, are well suited to decouple the control of chemical potentials. However, their use is less straightforward for the control of several tunnel couplings, which in addition to crosstalk typically exhibit a strongly nonlinear dependence on gate voltages. To tune a single coupling parameter at a time, van Diepen *et al.* presented a gradient descent procedure that incremented the virtual gate associated with the target parameter until the desired value was reached.¹⁷

Here, we propose and experimentally demonstrate an algorithm for the fine-tuning of qubits based on gate-defined quantum dots, which exploits machine learning for improving the efficiency of the tuning procedure. Our algorithm combines a gradient-based optimization with an adapted implementation of a Kalman filter. The latter allows efficient tracking of the gradients of the parameters in the multidimensional voltage space. Each measurement of the parameters at a new point in the voltage space is compared with the previous measurement and used to update the gradient. This approach results in a full automation of the simultaneous tuning of several coupled parameters. When applied to two tunnel couplings while keeping two chemical potentials fixed, it requires only 3–5 iterations to change a tunnel coupling by a factor 2, corresponding to 10–15 min of lab-time for the parameter extraction procedures currently used.

For the experimental realization, we use a double quantum dot in an AlGaAs/GaAs heterostructure in the same experimental setup as Botzem *et al.*¹⁶ and build on the parameter-extraction procedures developed there. The gate layout is shown in Fig. 1. The double quantum dot is designed to be used as a singlet-triplet qubit and features a neighboring sensing-dot for qubit readout based on spin-to-charge conversion. All measurements are based on RF-reflectometry and performed in a dilution refrigerator.

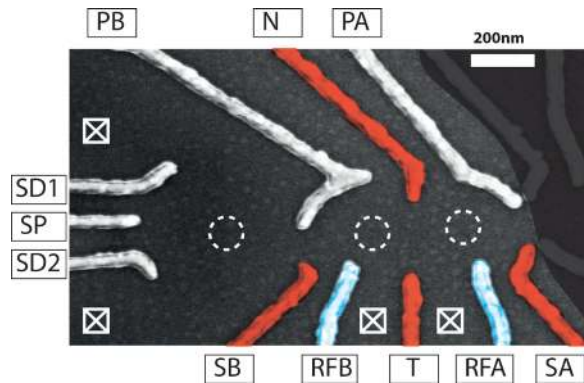


FIG. 1. Gate Layout of the sample used for the experimental demonstration. The dashed circles mark the approximate positions of quantum dots. Ohmic contacts are marked by crossed squares. The left dot is used as the sensing dot controlled by SD1, SD2, and SP. The RF-gates RFA and RFB, marked in blue, are utilized for rapid control of the chemical potential in the double quantum dot, which is statically set with the plunger gates PA and PB. The gates marked in red are used for tuning the tunnel couplings. Gates T and N are designed to control the inter-dot tunnel coupling, whereas SA and SB are meant to control the tunnel coupling to the electron reservoir.

Although there are 9 DC-gates defining the quantum dots, only 4 of them are used to fine-tune the parameters, namely, the gates N, T, SA, and SB. The voltages on SD1, SP, and SD2 are used to control the sensing dot, and the chemical potentials are controlled with PA and PB. The scans used to extract the parameters of interest are performed with the RF-gates RFA and RFB, which are DC-coupled to an arbitrary waveform generator. The qubit parameters that we want to tune are the strength of the inter-dot tunneling, characterized by the width of the inter-dot transition in gate voltage space, w ,^{16,18} and the time required to reload a singlet t_{sr} . The latter characterizes the tunnel coupling between one of the dots and its neighboring electron reservoir and is measured via the dependence of the load efficiency on the corresponding waiting time in the reload operation. Tuning these two parameters is sufficient to obtain a fully operational singlet-triplet qubit.¹⁶ The tunnel coupling to the other lead is almost closed to allow for latching readout.^{19–21}

Fine-tuning w and t_{sr} to some target values w^* and t_{sr}^* is complicated by the fact that w and t_{sr} are non-linear functions of four different voltages. To optimize their values, we employ Newton’s method with an approximated Jacobian, making our algorithm a quasi-Newton method. In the k th iteration of the algorithm, the gate voltages are updated according to the following formula:

$$\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} - (\mathbf{J}^{(k)})^{-1} \Delta^{(k)}, \quad (1)$$

where $\mathbf{v}^{(k)} = (v_N, v_{SA}, v_{SB}, v_T)^T$ is a vector containing the voltage configuration at step k , $\Delta^{(k)} = (t_{sr}^{(k)} - t_{sr}^*, w^{(k)} - w^*)^T$ contains the distance of the measured parameter values at step k from the target ones, and $\mathbf{J}^{(k)}$ is the Jacobian, with $J_{ij}^{(k)} = \partial p_i^{(k)} / \partial v_j^{(k)}$ for $j \in \{N, SA, SB, T\}$ and $p_i \in \{w, t_{sr}\}$. $(\mathbf{J}^{(k)})^{-1} \Delta^{(k)}$ is to be understood as the solution of the corresponding, possibly underdetermined system of linear equations with minimal Euclidean norm.

Our tuning procedure combines the Newton step with the Kalman filter as shown in the Unified Modeling Language (UML) diagram in Fig. 2. Each iteration begins with a new measurement of the parameters. If the measured values are already in the desired

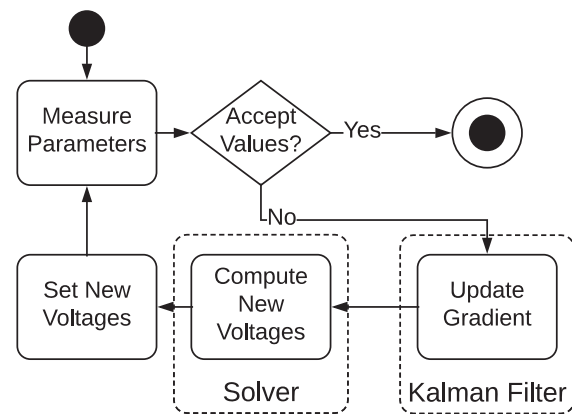


FIG. 2. UML activity diagram of the tuning algorithm. The solver can be any gradient based optimization algorithm. The Kalman update consists of Eqs. (2) and (3). Setting new voltages comprises a compensation of the shift in chemical potential by plunger gates.

parameter range, the voltages are accepted as the final state and the algorithm terminates. Otherwise, gradient updates are performed by a Kalman filter. The new gradients are then used in Eq. (1) to calculate the new voltages to be applied to the gates. The algorithm mimics the learning process of a human operator, who changes the voltages according to a certain expectation of how this will affect the qubit parameters, performs measurements to verify the result, and uses the information gained from these measurements to refine the understanding of the behavior of the system.

Since Eq. (1) is based on a linearization of the dependence of w and t_{sr} on the applied voltages, we restrict the maximal voltage change to 10 mV because the linear approximation by the gradient becomes less accurate when the steps are larger. On the other hand, the restriction should be chosen as large as possible because the ratio of the physical changes to the fluctuations of the parameters due to noise and disorder grows with the step size. In addition, smaller steps will lead to a larger number of steps for substantial changes. After the new voltages are set, the contrast in the sensing dot is optimized with the gates SD1 and SD2 and the chemical potential is corrected with the plunger gates PA and PB (see [supplementary material](#)).

Since the parameters are in general strongly nonlinear functions of the gate voltages, a naive implementation of a gradient-based optimization algorithm would require the Jacobian $\mathbf{J}^{(k)}$ to be remeasured by finite differences in every iteration. This is a time-consuming operation because it requires many measurements at different voltages. To avoid this issue, our algorithm takes a machine-learning approach and uses a Kalman filter^{22,23} to estimate the Jacobian at step k using the knowledge of $\mathbf{J}^{(k-1)}$ and the information drawn from a single set of measurements.

The Kalman filter is an algorithm designed to estimate a system of normally distributed random variables obtained from noisy measurements at discrete steps k and knowledge of their dynamics. We use an adapted version of the Kalman filter specific to our needs, which are the estimation of the gradient $\mathbf{g}_p^{(k)} = \langle \partial p^{(k)} / \partial v_j \rangle$ and the corresponding covariance matrix $C_{ij}^{(k)} = \langle (\partial p^{(k)} / \partial v_i - g_i^{(k)}) (\partial p^{(k)} / \partial v_j - g_j^{(k)}) \rangle$ of a parameter p in iteration k as a function of the control voltages v_{ij} for $i, j \in \{N, SA, SB, T\}$. Expectation values $\langle \cdot \rangle$ refer to the distribution of the uncertain parameters being tracked. For each $p \in \{w, t_{sr}\}$, we use a Kalman filter to track the estimation of its gradient described by \mathbf{g} and \mathbf{C} . Each instance of the Kalman filter approximates a row in the Jacobian \mathbf{J} by means of its distribution. The initial values $\mathbf{g}^{(0)}$ and $\mathbf{C}^{(0)}$ are measured by finite differences as discussed in the [supplementary material](#).

In each iteration, the Kalman filter uses the information gained from a new measurement of the parameters $p^{(k)}$ to update the values of \mathbf{g} and \mathbf{C} according to the following formulas:²³

$$\mathbf{g}^{(k)} = \mathbf{g}^{(k-1)} + \mathbf{K}^{(k)} (z^{(k)} - \mathbf{H}^{(k)} \mathbf{g}^{(k-1)}), \quad (2)$$

$$\mathbf{C}^{(k)} = (\mathbf{I} - \mathbf{K}^{(k)} \mathbf{H}^{(k)}) \mathbf{C}^{(k-1)} + \mathbf{Q}. \quad (3)$$

Here, $\mathbf{H}^{(k)}$ is an observation model (consisting in our application of 1×4 matrices) that maps the “state space” of the Kalman filter (i.e., the space of $\mathbf{g}^{(k)}$) onto the measurement space (i.e., the space of $p^{(k)}$), with elements $H_{ij}^{(k)} = v_j^{(k)} - v_j^{(k-1)}$. The product $\mathbf{H}^{(k)} \mathbf{g}^{(k-1)}$ then represents the predicted change in the parameter p due to the change in \mathbf{v} at step k , which is compared in Eq. (2) with the measured change $z^{(k)} = p^{(k)} - p^{(k-1)}$. The matrix $\mathbf{K}^{(k)}$ is the so-called Kalman gain

$$\mathbf{K}^{(k)} = \frac{\mathbf{C}^{(k-1)} \mathbf{H}^{(k),T}}{\mathbf{H}^{(k)} \mathbf{C}^{(k-1)} \mathbf{H}^{(k),T} + \Delta z^{(k)^2}}, \quad (4)$$

which depends on both the uncertainty of our knowledge of the gradients represented by the covariance matrix $\mathbf{C}^{(k)}$ and the measurement uncertainty $\Delta z^{(k)^2} = \delta p^{(k)^2} + \delta p^{(k-1)^2}$, with $\delta p^{(k)}$ being the error on the measurement of $p^{(k)}$ (see [supplementary material](#)). An inaccurate measurement has a very large $\Delta z^{(k)^2}$ and therefore a small gain. The information gained with the measurement not only contributes to updating the value of the gradient [see Eq. (2)] but also reduces the covariance matrix $\mathbf{C}^{(k)}$ by a factor determined by the Kalman gain [first term in Eq. (3)]. The algorithm becomes Broyden’s method in the limit $\Delta z \rightarrow 0$. In addition, we include a fixed increase by the term \mathbf{Q} , which accounts for the additional uncertainty related to our lack of knowledge of how the gradient changes while changing voltages, i.e., of how $\mathbf{g}^{(k)}$ deviates from $\mathbf{g}^{(k-1)}$.

In our case, \mathbf{Q} is chosen heuristically with the constraint that the estimated increase in the uncertainty of the parameter evaluation, $\mathbf{H}^{(k)} \mathbf{Q} \mathbf{H}^{(k),T}$, must be of the same order of magnitude as the measurement uncertainty $\Delta z^{(k)^2}$ for typical values of $\mathbf{H}^{(k)}$. This ensures a reasonable Kalman gain $\mathbf{K}^{(k)}$ [compare Eq. (4)] and hence the change in the prediction $\mathbf{g}^{(k)}$. Note that too small values for \mathbf{Q} lead to an almost constant \mathbf{g} , which can cause slow convergence or oscillatory behavior of Eq. (1). Too large values on the other hand lead to abrupt changes in \mathbf{g} , which may be problematic if individual measurements occasionally give wrong results, e.g., because the underlying fit does not converge (see [supplementary material](#)).

To test the algorithm, we cyclically changed the target parameters (t_{sr}, w) in the sequence (20 ns, 270 μV)–(70 ns, 270 μV)–(70 ns, 180 μV)–(20 ns, 180 μV)–(20 ns, 270 μV). For each pair of set points, we performed 21 iterations regardless of the convergence. The resulting voltage changes are not necessarily cyclical as the system is underdetermined. The choice of the lower set point of t_{sr} is limited by the bandwidth of the data acquisition hardware, and the lower set point of w is limited by temperature broadening. We chose the upper limits such that we restrict the range to values typically used in experiments. The observed fluctuations around the set points can largely be attributed to noise in the parameter extraction. They are the main limiting factor for the achievable accuracy of the tuning result. According to our experience, this level of accuracy is sufficient for qubit operations, which can tolerate static variations of the parameters considered here by at least 50%. The data in Fig. 3(a) demonstrate that the parameters can be tuned individually within three to five steps with a total duration of 10–15 min. This time includes the compensation of shifts in the chemical potentials and the adjustment of sensor operating points. It is determined by the measurement time, while the computational time is negligible. This performance is comparable to that reported in Ref. 17 for the tuning of a single tunnel coupling. A speedup of the tuning routine could be achieved by optimizing the way we extract w and t_{sr} . The time required by each measurement is discussed in the [supplementary material](#).

The estimates of the gradients $\mathbf{g}^{(k)}$ are plotted in Fig. 3(b). We see that abrupt changes in the gradient estimates mostly occur in the first steps of each region right after set points are changed. In the upper graph in Fig. 3(b), the absolute values of the elements $\partial t_{sr} / \partial V_{SB}$ and $\partial t_{sr} / \partial V_T$ are much larger than those of the other two elements because they form the tunnel barrier to the lead next to

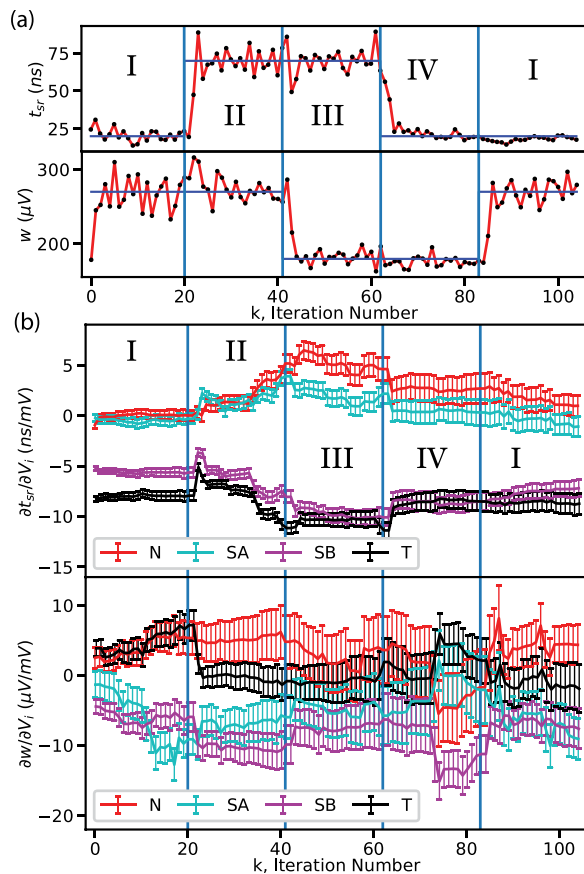


FIG. 3. Experimental demonstration of the tuning algorithm. The iteration number k is given by the number of cycles in the diagram shown in Fig. 2. (a) The parameters are plotted in red. The horizontal blue lines are the current set points, and vertical blue lines mark changes in the set point. Different set points are counted by roman numerals. The parameters are tuned within 3–5 iterations into the desired range with an accuracy limited by parameter fluctuations due to statistical noise in the measurement. (b) Gradients of the parameters with respect to the voltages on the gates N, SA, SB, and T, tracked by the Kalman filter. The error bars are the square roots of the diagonal elements in the gradient’s covariance matrix, which are larger in the lower graph because the measurements of the inter-dot transition are generally less accurate. The gradients in the regions with set point I show some differences although the parameters are similar. This can be explained by the influence of the initial gradient estimation and different gate voltages as the system is underdetermined with two parameters controlled by four voltages. (Gate voltages and extension in the [supplementary material](#)).

RFB (see Fig. 1), which is used for the electron exchange in the singlet reload mechanism. The gradient elements $\partial t_{sr}/\partial v_j$ for $j \in (N, SA, SB, T)$ are almost constant, indicating only weak effects due to non-linearity.

The gradient of the transition width in the lower graph in Fig. 3(b) is more complicated, presumably because all gates control the positions of the quantum dots and hence the inter-dot tunnel coupling and the transition width. The element $\partial w/\partial V_{SB}$ changes by a factor of 3, and the other elements change their sign during the experimental demonstration. These changes in the gradients confirm that the dependence of the qubit parameters on the control voltages is strongly

non-linear, which indicates that tuning procedures based on pre-calibrated gradients would be inefficient and underpins the advantage of tracking the changes of the gradients during the tuning procedure.

To facilitate the adoption of this approach, we provide an implementation of the algorithm as a python package named *qtune*.²⁴ It contains our implementation of the Kalman filter and more tools that simplify an automated fine-tuning program. The implementation complies with good software engineering practices by including a full documentation and unit tests with high coverage. A general interface makes the package adaptable to other setups.

In conclusion, we used the Kalman filter to construct a fully automated fine-tuning procedure. Thereby, we demonstrated the ability of the Kalman filter to be used in combination with a gradient based optimization algorithm to efficiently solve a non-linear optimization problem without re-measuring the gradient. Thus, the algorithm does not only save time and resources but also provide valuable information about the qubit in the form of the gradient of its characteristics as a function of voltages, which can be used for evaluating its tunability.

Improvements of the convergence behavior could likely be achieved by choosing Q depending on the size of the voltage step in each iteration. In comparison with other quasi-Newton methods like Broyden’s methods, our procedure offers the advantage of taking statistical errors into account. The resulting performance advantage is yet to be demonstrated by detailed benchmarks. An additional performance increase may be achievable by feeding the logarithms of tunnel couplings into the algorithm, thus linearizing the expected exponential dependence of the tunnel coupling on gate voltages.²⁵ However, this would likely not capture all non-linearities, with effects of shifting the electron locations being one potential counterexample.

The algorithm has been tested on a semiconductor spin qubit in a AlGaAs/GaAs double quantum dot but could also be applied to other types of qubits, including any qubit based on gate defined quantum dots. Automated tuning procedures not only will be needed for operating quantum processors but can also be very valuable for the systematic characterization and optimization of qubit technology and reproducibility. In fact, the possibility to tune a qubit is a key criterion for its functionality and is intimately related to the tuning procedure employed. Hence, “smart” algorithms can make any given qubit design more successful.

See [supplementary material](#) for an extension of the measurement data in Fig. 3 and the corresponding voltage values of the full experimental demonstration for completeness. We also elaborate on the control of the sensing dot and the chemical potentials in order to discuss the time consumption of our algorithm in detail. We include a description of the initial values, our method to estimate the measurement uncertainty, and the choice of Q in Eq. (2) as well.

We thank F. Haupt for helpful input on this article and Robert P. G. McNeil for the fabrication of the sample. We acknowledge support from the Impulse and Networking Fund of the Helmholtz association, the Helmholtz Nano Facility (HNF) at the Forschungszentrum Jülich,²⁶ the Deutsche Forschungsgesellschaft under grant BL 1197/2-1 and BL 1197/4-1, and the Excellence Initiative of the German federal and state governments. A. Ludwig and A. D. Wieck gratefully acknowledge support of DFG-TRR160, BMBF - Q.Link.X 16KIS0867, and DFH/UFA CDF A-05-06.

REFERENCES

- ¹L. M. K. Vandersypen, H. Bluhm, J. S. Clarke, A. S. Dzurak, R. Ishihara, A. Morello, D. J. Reilly, L. R. Schreiber, and M. Veldhorst, *npj Quantum Inf.* **3**, 34 (2017).
- ²D. M. Zajac, A. J. Sigillito, M. Russ, F. Borjans, J. M. Taylor, G. Burkard, and J. R. Petta, *Science* **359**, 439 (2018).
- ³T. F. Watson, S. G. Philips, E. Kawakami, D. R. Ward, P. Scarlino, M. Veldhorst, D. E. Savage, M. G. Lagally, M. Friesen, S. N. Coppersmith, M. A. Eriksson, and L. M. Vandersypen, *Nature* **555**, 633 (2018).
- ⁴M. Veldhorst, C. H. Yang, J. C. Hwang, W. Huang, J. P. Dehollain, J. T. Muhonen, S. Simmons, A. Laucht, F. E. Hudson, K. M. Itoh, A. Morello, and A. S. Dzurak, *Nature* **526**, 410 (2015).
- ⁵J. Yoneda, K. Takeda, T. Ostuka, T. Nakajima, M. R. Delbecq, and G. Allison, *Nat. Nanotechnol.* **13**, 102 (2018).
- ⁶P.-A. Mortemousque, E. Chanrion, B. Jadot, H. Flentje, A. Ludwig, A. D. Wieck, M. Urdampilleta, C. Bauerle, and T. Meunier, preprint [arXiv:1808.06180](https://arxiv.org/abs/1808.06180) (2018).
- ⁷C. Volk, A. M. J. Zwerver, U. Mukhopadhyay, P. T. Eendebak, C. J. van Diepen, J. P. Dehollain, T. Hensgens, T. Fujita, C. Reichl, W. Wegscheider, and L. M. K. Vandersypen, preprint [arXiv:1901.00426](https://arxiv.org/abs/1901.00426) (2019).
- ⁸A. R. Mills, D. M. Zajac, M. J. Gullans, F. J. Schupp, T. M. Hazard, and J. R. Petta, *Nature Commun.* **10**, 1063 (2019).
- ⁹T. Fujita, T. A. Baart, C. Reichl, W. Wegscheider, and V. L. M. K., *npj Quantum Inf.* **3**, 22 (2017).
- ¹⁰N. Samkharadze, G. Zheng, N. Kalhor, D. Brousse, A. Sammak, U. C. Mendes, A. Blais, G. Scappucci, and L. M. Vandersypen, *Science* **359**, 1123 (2018).
- ¹¹R. Hanson, L. P. Kouwenhoven, J. R. Petta, S. Tarucha, and L. M. K. Vandersypen, *Rev. Mod. Phys.* **79**, 1217 (2007).
- ¹²W. G. van der Wiel, S. De Franceschi, J. M. Elzerman, T. Fujisawa, S. Tarucha, and L. P. Kouwenhoven, *Rev. Mod. Phys.* **75**, 1 (2002).
- ¹³T. A. Baart, P. T. Eendebak, C. Reichl, W. Wegscheider, and L. M. K. Vandersypen, *Appl. Phys. Lett.* **108**, 213104 (2016).
- ¹⁴S. S. Kalantre, J. P. Zwolak, S. Ragole, X. Wu, N. M. Zimmerman, M. D. Stewart, and J. M. Taylor, *npj Quantum Inf.* **5**, 6 (2019).
- ¹⁵S. Foletti, H. Bluhm, D. Mahalu, V. Umansky, and A. Yacoby, *Nat. Phys.* **5**, 903 (2009); e-print [arXiv:1009.5343v1](https://arxiv.org/abs/1009.5343v1).
- ¹⁶T. Botzem, M. D. Shulman, S. Foletti, S. P. Harvey, O. E. Dial, P. Bethke, P. Cerfontaine, R. P. G. McNeil, D. Mahalu, V. Umansky, A. Ludwig, A. Wieck, D. Schuh, D. Bougeard, A. Yacoby, and H. Bluhm, *Phys. Rev. Appl.* **10**, 054026 (2018).
- ¹⁷C. J. van Diepen, P. T. Eendebak, B. T. Buijtdorp, U. Mukhopadhyay, T. Fujita, C. Reichl, W. Wegscheider, and L. M. K. Vandersypen, *Appl. Phys. Lett.* **113**, 033101 (2018).
- ¹⁸L. DiCarlo, H. J. Lynch, A. C. Johnson, L. I. Childress, K. Crockett, C. M. Marcus, M. P. Hanson, and A. C. Gossard, *Phys. Rev. Lett.* **92**, 226801 (2004).
- ¹⁹S. A. Studenikin, J. Thorgrimson, G. C. Aers, A. Kam, P. Zawadzki, Z. R. Wasilewski, A. Bogan, and A. S. Sachrajda, *Appl. Phys. Lett.* **101**, 233101 (2012).
- ²⁰J. D. Mason, S. A. Studenikin, A. Kam, Z. R. Wasilewski, A. S. Sachrajda, and J. B. Kycia, *Phys. Rev. B* **92**, 125434 (2015).
- ²¹P. Harvey-Collard, B. D'Anjou, M. Rudolph, N. T. Jacobson, J. Dominguez, G. A. Ten Eyck, J. R. Wendt, T. Pluym, M. P. Lilly, W. A. Coish, M. Pioro-Ladrière, and M. S. Carroll, *Phys. Rev. X* **8**, 021046 (2018).
- ²²R. E. Kalman, *J. Basic Eng.* **82**, 35 (1960).
- ²³G. Welch and G. Bishop, *An Introduction to the Kalman Filter, Technical Report* (Chapel Hill, NC, USA, 1995).
- ²⁴J. Teske and S. Humpohl, <https://github.com/qutech/qtune> for “qtune Fine-Tuning Package.”
- ²⁵C. B. Simmons, M. Thalakulam, B. M. Rosemeyer, B. J. Van Bael, E. K. Sackmann, D. E. Savage, M. G. Lagally, R. Joynt, M. Friesen, S. N. Coppersmith, and M. A. Eriksson, *Nano Lett.* **9**, 3234 (2009).
- ²⁶W. Albrecht, J. Moers, and B. Hermanns, *J. Large-Scale Res. Facil.* **3**, A112 (2017).