Quantum Science and Technology

OPEN ACCESS

RECEIVED
1 June 2020

ACCEPTED FOR PUBLICATION
1 June 2020

PUBLISHED
23 June 2020

ERRATUM

Erratum: Randomized benchmarking in the analogue setting (2020 *Quantum Sci. Technol.* **5** 034001)

E Derbyshire[1,4], **J Yago Malo**[2], **A J Daley**[2], **E Kashefi**[1,3] **and P Wallden**[1]

[1] School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, United Kingdom
[2] Department of Physics and SUPA, University of Strathclyde, Glasgow G4 0NG, Scotland, United Kingdom
[3] Laboratoire d'Informatique de Paris 6, CNRS, Sorbonne Université, 4 place Jussieu, 75005 Paris, France
[4] Author to whom any correspondence should be addressed.

E-mail: e.e.derbyshire@sms.ed.ac.uk

Due to an error during the publishing process the below errors occurred.
Algorithm 1 should be labelled as algorithm 2. It should appear as:

Algorithm 2. Analogue randomized benchmarking.

1: Sample uniformly from $\{U_k\}$ a number of sequence time-lengths S_T and run a sequence S_η of time-length $T \in S_T$ where: $S_\eta = \left[\Lambda_{U_{k_\eta}}, \ldots, \Lambda_{U_{k_\eta}}, \Lambda^\dagger_{U_{k_\eta}}, \ldots, \Lambda^\dagger_{U_{k_1}}\right]$ on your system such that if each unitary was perfectly implemented your system would be returned to initial state ρ_φ. Here we systematically invert each preceding unitary.
2: Repeat the sequence R times; record the survival probability $P_\eta = \mathrm{Tr}\left[E_\psi S_\eta\left(\rho_\psi\right)\right]$ for this sequence.
3: Repeat the above for various sequences of the same time-length to get the average survival probability for each sequence time-length: $P_T = \left[E_\psi S_T\left(\rho_\psi\right)\right]$ where S_T is the average over all sequences of time-length, T.
4: Repeat the above steps for sequences of different time-lengths and plot the average survival probability against time-length P_T vs T.
5: Fit the results to a predetermined decay curve of the following or similar form: $P_T = A + Bf^T$, where again T is the sequence time-length and f represents the fidelity decay parameter of the process, with A and B fit parameters that absorb SPAM errors. And again, the average error rate is characterised by r where $r = (d-1)\left(1-f\right)/d$ and d is the dimension of the Hilbert space for a system of qubit size n $(d = 2^n)$.

This algorithm should have appeared in section 3.3. It is cited on page 6: in this section, we first give an overview of extending RB to the analogue setting, with technical details from section 3.1 onwards, and our ARB protocol (see algorithm 2) in section 3.1.

Algorithm 2 should be labelled as algorithm 1. It should appear as:

Algorithm 1. Digital randomized benchmarking.

1: Sample uniformly from $\{U_k\}$ a number of sequence lengths S_l and run a sequence S_η at length $l \in S_1$ where: $S_\eta = \Lambda_{U_{k_\eta}} + \Lambda_{U_{k_1}}, \ldots, \Lambda_{U_{k_\eta}}$, and $\Lambda_{U_{k_{\eta+1}}}$ is a single operator deterministically chosen to invert the preceding sequence of unitaries (i.e. $\Lambda_{U_{k_{\eta+1}}} = \left[\Lambda_{U_{k_1}}, \ldots, \Lambda_{U_{k_\eta}}\right]^\dagger$). This sequence should return the system to its initial state ρ_ψ.
2: Repeat this sequence R times and record $\mathrm{Tr}\left[E_\psi S_\eta\left(\rho_\psi\right)\right]$ to see if initial state ρ_ψ survived the sequence S_η and call this the survival probability P_η for sequence η.
3: Repeat this for varying sequences of the same length l and find the average probability that the initial state survived for this sequence length $\mathrm{Tr}\left[E_\psi, S_l\left(\rho_\psi\right)\right]$ where S_l represent the average over all sequences of length l. Call this the average probability for length l: P_l.
4: Repeat the above steps for sequences of different lengths, and plot average survival probability against sequence length, i.e. P_l vs l. Fit results to a pre-determined decay curve: $P_l = A + Bf^1$ where l is the sequence length and f is the fidelity decay parameter, with A and B absorbing SPAM errors.
5: The average error rate can be charactterised by r where $r = (d-1)\left(1-f\right)/d$, and d is the dimension of the Hilbert space for a system of qubit size $n = (d = 2^n)$.

This algorithm should have appeared in section 2. It is cited on page 3: we present here a basic randomized benchmarking protocol (see algorithm 1), for completeness, and for technical details and further explanation of how the method works we refer to appendix A.

It is also cited on page 6: the standard form of RB (see algorithm 1) involves a single deterministically chosen inversion operator ΛU_{k+1} that inverts the preceding unitary sequence.

It is also cited on page 7: the parameters for the standard RB protocol (see algorithm 1 in section 2) still apply, but we redefine the form of S_l and introduce another measure for the sequence length: S_T where $S_T \equiv S_l$ and T represents the total time to run each sequence of length l, i.e. $T = \mathrm{d}t \cdot l$ and so, we highlight that the average error-rate gained from this protocol is the average error as a function of physical time T.

In addition the Orcid ID of author J Yago Malo should be included. This ID is: 0000-0001-5588-9183.