Investigating student use of a flexible tool for simulating and visualizing quantum mechanics

C. A. Weidner,[1,*] S. Z. Ahmed,[1,*] J. H. M. Jensen,[1] J. F. Sherson,[1] and H. J. Lewandowski[2,3]

[1]*Department of Physics and Astronomy, Aarhus University, 8000 Aarhus C, Denmark*
[2]*Department of Physics, University of Colorado, Boulder, CO 80309, USA*
[3]*JILA, National Institute of Standards and Technology and University of Colorado, Boulder, CO, 80309, USA*

As education researchers gain a broader understanding of how students learn quantum mechanics, new pedagogical and technical resources are being developed to facilitate student learning. To further research-based knowledge of student learning of quantum mechanics, we present a study on the use of Quantum Composer, a flexible, flow-based tool for the exploration and simulation of quantum mechanical systems in one dimension. To explore Composer's impact on students' knowledge of quantum mechanics, we carried out think-aloud interviews where students worked through an exercise exploring the statics and time-dynamics of quantum states in single and double harmonic well potentials. Student *Outcomes* are then cross-coded with their observed *Interactions* with Composer. We find that defined *Outcomes* of *Recollection*, *Reinforcement* and *Discovery* happen most often when students are using the Composer tools that allow them to visualize quantum states, simulate their time dynamics, and change parameters repeatedly in order to understand how systems are represented in both the static and dynamic cases.

I. INTRODUCTION

There have been several studies identifying challenges students have with building a conceptual understanding of quantum mechanics at both the undergraduate and graduate level [1–11]. For example, it has been shown that students face difficulties in sketching the shape of a wavefunction even if they know the correct mathematical form [2]. Students also struggle with correctly describing the time evolution of a non-stationary state [2, 3, 12] and the real and imaginary parts of a stationary state [13]. Thus, tools that allow students to visualize and simulate quantum phenomena may foster better student understanding of challenging concepts in quantum mechanics.

A number of tools have been developed for quantum visualization [14–19], including a wide array of simulations, among them the PhET [20], QuVIS [21], and QuILT [22] simulations. Some of these tools have been used to study aspects of student understanding of quantum mechanics, including wavefunction time-dependence [13], tunneling [23], and expectation values [24]. These simulations are modular in that a specific tool typically addresses a small subset of concepts, and a curriculum is built around the tools to foster student understanding. One of the broadest research questions one can ask of such a simulation and visualization tool is simply: does use of the tool affect student reasoning regarding quantum mechanical concepts, and if so, how?

Here, we explore these questions as they relate to Quantum Composer [25], a node- and flow-based quantum visualization and simulation tool based on the QEngine developed at Aarhus University for quantum research and control [26]. In contrast to modular simulations, Composer uses a flexible, drag-and-drop interface allowing students and instructors to build or modify simulations (called *flowscenes*) from the ground-up without any prior programming knowledge. With Composer, a variety of one-dimensional quantum scenarios can be simulated, and the statics and dynamics of problems can be explored through a large range of visualization tools [27]. As this is a new and unstudied tool, we aim to first explore broadly how students use it to reason about canonical quantum mechanical systems. To accomplish this goal, we use think-aloud interviews of students working through an exercise using Quantum Composer. We find that when students are presented with a pre-made simulation in Composer and surrounding exercise materials, they are able to use the visualization tools contained in the simulation to help build and reinforce their understanding of a problem.

II. METHODOLOGY

Research Context This study took place at Aarhus University (AU), the second-largest university in Denmark, with approximately 38,000 students enrolled in Bachelor's, Master's, and PhD programs. The student population is predominantly white and native to Denmark. For this study, we recruited students who were enrolled in a second-year undergraduate quantum mechanics for nanoscientists course, based on Griffiths' textbook [28]. Nanoscience students typically take courses covering mechanics, thermodynamics, calculus, electrodynamics, waves, and optics before the quantum course. In addition, both students in this study also reported having had some introductory quantum mechanics in a physical chemistry course taken concurrently with the quantum course. The quantum course was taught in English, and the study was in English, but all of the students who volunteered for our study spoke English as a second language.

The quantum course had seven theoretical exercises (homework assignments) based on Griffiths problems. Several of the problems required students to use programming tools like MATLAB. Three times throughout the course, the students were given additional exercises built around Composer flowscenes (see Fig.1). None of the exercises were graded. Students were provided with relevant flowscenes, and the exercises were designed so that students did not have to build anything from scratch (although they were shown how this could be done). The first set of exercises explored the infinite and finite square well, the second set investigated superposition and expectation values, and the third discussed time-independent perturbation theory. None of the Composer exercises explored time dependence, but time-dependence was explored in the course. Before the first exercise, students were given a 15-minute presentation on how to use Composer.

Think-aloud Interviews. Based on the coverage of concepts in the course, we designed our study around scenarios that addressed quantum states and their time-evolution for the particular cases of single- and double-well potentials represented by a harmonic well and two split harmonic wells, respectively. We chose these scenarios because they represented a system that students had seen in their course (the single well) and a second system that extended beyond what they had learned in class (the double well).

Students were recruited in-class and via messages posted on the course's online message board, and volunteers were asked to give two hours of their time. Participants were compensated with a gift card worth 500 Danish kroner. The think-aloud interviews took place in the weeks following the final exam, and all interviews were conducted by S.Z.A. Students were given one hour to complete the exercise, after which they were asked some basic questions on their background, thoughts about Composer, and demographic information. Participants were told that a response to the demographic questions was optional. Due to the small number of participants, we do not report demographics here as they could lead to identification of the students.

Five of the 31 students in the course participated in the think-aloud interviews, but we present results from the initial analysis of only two students here. In these interviews, the exercise and think-aloud format was briefly explained by the interviewer, but after this explanation, the interviewer interacted with the student only to prompt the student to think aloud and to offer help on Composer-specific issues (e.g.,

* These two authors contributed equally.

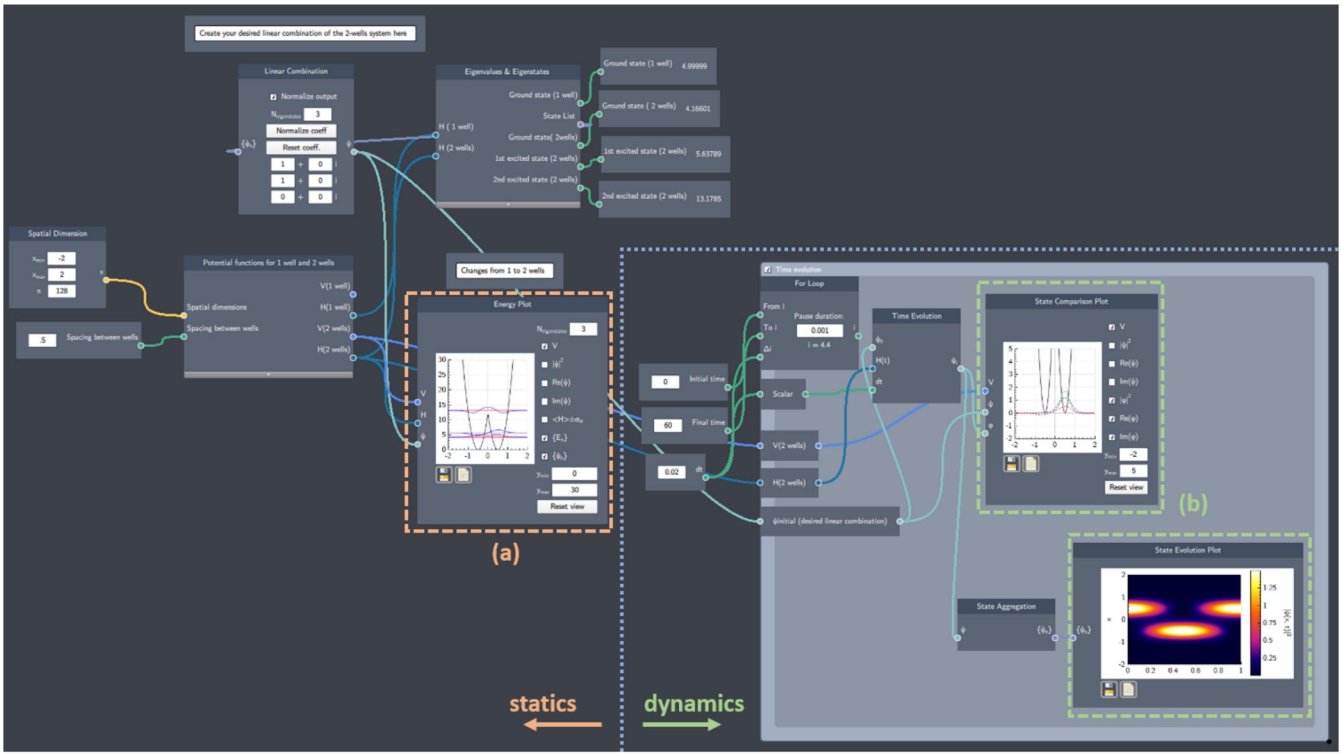

FIG. 1. A screenshot of the second flowscene used by the interview participants in Scenario 2: The double-well potential. The blue dotted lines separate the time-independent (static) part of the simulation from the time-dependent dynamics that take place in the *Time evolution* loop. Students can change all of the parameters shown in white boxes within separate nodes. Plots that students could use for visualization are shown for (a) the time-independent case (orange, dashed box) and (b) the time-dependent case (green, dashed box). The plots in (b) updated continuously if the student initiated time evolution. The flowscene for Scenario 1 was similar to what is shown here.

how to play time evolution). Students were also provided with a reference sheet that described the relevant nodes in Composer and brief descriptions of the different plots used in the simulations (cf. Fig. 1).

During the interviews, students' voices and computer screens were recorded. The interviewer also kept any sketches or notes made by the student during the interview, and these were used for coding during the data coding portion of the study.

The exercise covered, in order, the following aspects of the two scenarios described above:

- **Scenario 1: Single-well potential**
  - Static eigenstates
  - Time dynamics of single eigenstates
  - Building superposition states
  - Time dynamics of superposition states
- **Scenario 2: Double-well potential**
  - Eigenvalues as a function of well separation
  - Eigenstates as a function of well separation
  - Using superposition to localize a state in each well.
  - Time evolution of this superposition state as a function of well separation

For the interviews, we provided previously constructed flowscenes for the participants to work with. To minimize confusion, we built a separate flowscene for each of the two scenarios, the latter of which is shown and described briefly in Fig. 1. The first scenario used a *Predict, Explore, Revise* framework. Here, students were asked to *Predict* (using either words, equations, or sketches, but without using Composer) how the system is represented statically or will behave under time evolution, depending on the part of the exercise they are working with. Afterwards, they would use Composer to *Explore* the relevant aspect of the exercise and use their explorations to *Revise* their understanding. These key words were also used in the second scenario, but, as the participants had not seen the double-well potential before, this part of the exercise started with exploration of the system behavior (cf. the first two points in the list above) before asking students to use the *Predict, Explore, Revise* framework for the last two points.

Interview Coding Scheme. The codebook was developed after the interviews were completed. There were three *a priori* code categories that aimed to capture the *Concepts* students used, their *Interactions* with Composer, and the *Outcomes* of these interactions. During the coding process, we added two code categories (*Tools/Representations* and *Understanding of Composer*) emergent from the data. Initially, the interviews (audio and video data, as well as any handwrit-

ten material) were coded independently by C.W. and S.Z.A in one-minute chunks. After multiple iterations failed to produce sufficient inter-rater reliability, the codebook was simplified to what is shown in Table I. The failure of the inter-rater reliability happened primarily for the *Concept* codes, as there was an ambiguity of the concept that the students were trying to convey due to their lack of fluency with the technical language of quantum mechanics. Hence, the definitions of the *Concept* codes were simplified and the interviews were collaboratively coded by C.W. and S.Z.A.and the coding was discussed with H.J.L. All discrepancies were resolved minute-by-minute via discussion.

TABLE I. The codebook used in the study presented in this work. In addition to the codes presented here, we coded for the phase of the *Predict, Explore, Revise* framework the student was working in, as well as the *Concepts* the exercise was asking the students to explore.

Code	Subcode (emergent)
Tool/Representation	Math
	Sketches
Understanding of Composer	Orientation
	Asking for and receiving help
Interaction	Set distance between wells
Setting quantities	Set number of eigenstates
	Set linear combination coefficients
	Changing time settings
Inspection	Play time evolution
	Visualization
	Reading eigenvalues
Iteration	
Outcomes	Recollection
	Reinforcement
	Discovery

In addition to the codes presented in the table, we coded for which phase of the *Predict, Explore, Revise* framework the student was working with at the time. During initial coding, we found that exploration and revision often happened without a clear delineation between the two cases, so these were coded together. In addition, we coded for which *Concepts* that served to distinguish which physical concepts the exercise was asking the students to reason about. These codes distinguished between exploration of time-independent and time-dependent systems, single-eigenstate and superposition systems, and single- vs. double-well systems. This was coded simply to determine how much time students were spending working with which aspects of the exercise. Typically, multiple *Concepts* were coded simultaneously.

Interviews were also coded based on the Composer-relevant actions interviewees were taking during the *Explore/Revise* portions of the exercise. This included codes relevant to the *Understanding of Composer*, such as when a student was trying to orient themselves within Composer or

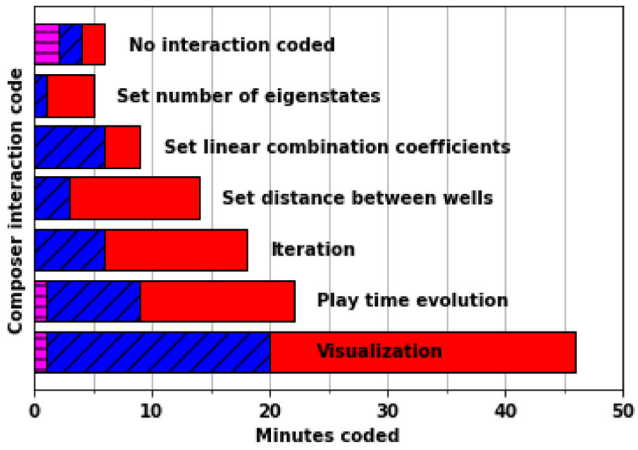

FIG. 2. Bar chart showing the number of minutes that each *Outcome* (*Recollection*: magenta horizontally-striped bars, *Reinforcement*: blue diagonally-striped bars, and *Discovery*: solid red bars) was coded alongside each of the *Interaction* codes, ordered by decreasing number of minutes coded per *Interaction*. For reference, the number of minutes where an *Outcome* was coded and no *Interactions* were coded simultaneously is shown on the top bar. The total number of minutes coded for each *Interaction* is shown in Table II. Codes with fewer than five total minutes coded are not shown here.

when they asked the interviewer for help. Additionally, we coded for *Interactions* wherein interviewees inspected values, plots, or observed dynamics in Composer. This includes playing time evolution, visualizing what was happening in the various plots in each scenario, and *Iteration*, where students changed parameters repeatedly in order to see how a parameter affected the simulation result. We also coded when interviewees set quantities in Composer, like the number of eigenstates shown in the static plot (Fig. 1(a)) or the distance between the wells in the double-well system.

Finally, we also coded three *Outcomes* that could occur during the *Explore/Revise* segments. The first of these is *Reinforcement*, which was coded when students noted an agreement between the results of working with Composer and their predictions in the *Predict* phase. In contrast, the *Recollection* and *Discovery Outcomes* were coded when students made incorrect predictions. The difference between these is that during *Recollection*, the students state that they recalled the correct relevant physics concepts from their previous knowledge (often gained from the course). We coded *Discovery* when students encountered a contradiction to their prediction and had to revise their understanding. For example, Interviewee 2 (I2) incorrectly predicted the shape of the imaginary part of the wavefunction (a common mistake made by students [13]) and had to revise their thinking when confronted with Composer visualizations. In addition, when students demonstrated evidence of new conceptual or mathematical understanding, this was also coded as *Discovery*. This arose, e.g., when students were exploring the unfamiliar double-well system.

III. RESULTS AND DISCUSSION

The results presented in this paper will focus on the how the coding of interviewee outcomes correlated with Composer interactions in order to elucidate how Composer impacted their reasoning. In Table II, we tabulate the total number of minutes coded per interview and in total. We find that the *Recollection* outcome was sparsely coded, meaning that students typically either reinforced previous understanding or learned something new from the exercise. Our analysis then looked for how the *Outcomes* correlated with Composer *Interactions*. A bar chart showing the number of minutes that each *Outcome* was coded simultaneously with the eight *Interaction* codes is shown in Fig. 2. We also show on this chart the number of minutes where an *Outcome* was coded but no *Interaction* was coded at the same time, showing that typically, *Outcomes* were coded simultaneously with *Interactions*. Additionally, the data show that *Discovery* and *Reinforcement* were coded with roughly equal frequency.

TABLE II. Total number of minutes coded per *Outcome* for each interview. Interviews 1 and 2 were coded for 50 and 66 minutes, respectively.

	Recollection	**Reinforcement**	**Discovery**
Interview 1	0	14	17
Interview 2	3	7	12
Total	3	21	29

We find that the bulk of the simultaneous coding results were focused on *Inspection* codes, where participants were actively using Composer's time evolution and visualization features. Additionally, *Outcomes* would often arise as students would iterate on various parameters, indicating that students' understanding might be facilitated by Composer's flexibility and ability to rapidly cycle through parameters. Interviewee 1 (I1) was able to reason through how the time dynamics of the real part, imaginary part, and probability density of a superposition state initially localized to one of the two wells changed over time through the use of all three of these *Inspection* behaviors. This is shown in the following quote, where the well spacing is shown in parentheses.

(0.75) It's slowly moving. It's moving fast up and down and then slowly moving towards the other well, and not as fluently (sic) *as before because of the potential barrier in between the two wells, but still moving over there and after some time resetting itself...then moving back again at the same pace...if I lower [the spacing between wells], it would probably move faster...(0.5)...Yep. Moving remarkably faster...if we tried to set it at one, it would still move...(1)...it's moving a little, just very, very, very slow. Okay, so the higher the spacing between the wells is, the harder it is for it to get to the other well...if we change this to 0.1, it should move back and forth almost in the [single-well] case from before. (0.1) Yep. Fluent* (sic) *motion back and forth, almost like with the one-well system.*

This segment was coded both as *Reinforcement* and *Discovery*, because while the student's predictions were correct, the student is also identifying that the behavior of the system is periodic in time, which is new understanding not identified in the *Predict* phase. The student is using the visualizations from Composer (Fig. 1(b)) to formulate an understanding of how a localized state tunnels between two wells.

In some cases, we found that students would use something they learned or recalled with the help of Composer (coded as *Discovery* or *Recollection*, respectively) to rethink an incorrect prediction and subsequently make a correct prediction later in the exercise. For example, I2 initially did not predict that the probability density of an eigenstate would be stationary in time. Using Composer helped them to recall what they had learned in class (coded as *Recollection*), as shown below.

Oh, I need the probability density...and here we go. Let's see. There's something wrong...why isn't that moving more?...Oh, okay. So this part, I thought earlier would be moving. [It] doesn't move because it's the probability. But it makes sense, actually, from the theory that I have learned.

Later in the exercise, when asked to predict the dynamics of the first excited state, I2 indicates a change in thinking, stating "Now I have learned that the probability density would also look the same [as time evolves]." This quote shows that I2 has developed an improved understanding in the earlier part of the exercise and applied this to make a correct prediction.

While this study cannot draw any generalizable conclusions about student learning, these results suggest that students can use their interactions in Composer to test, re-evaluate, and/or verify their reasoning. Indeed, I2 states:

From the program here, I can now connect it with some of the things I have learned earlier. I'm not that good at equation (sic)*...I'm more visualized* (sic)*, so I like the program. It makes more sense. And then afterwards I normally learn the equation.*

IV. CONCLUSIONS

In this work, we have shown how Quantum Composer can be applied to a comprehensive exercise that covers a variety of topics. Our data show that student *Outcomes* are largely tied to Composer interactions, specifically those in which students use Composer to inspect and explore a system. Therefore, these data support the hypothesis that student reasoning regarding quantum mechanical phenomena is supported by the visualizations facilitated by Composer. Future work will focus on bolstering these initial conclusions with data from three more interviews. In addition, we will explore more in-depth cross-coding between *Outcomes*, *Interactions*, and *Concepts*. With this, we hope to broaden the scope of Composer's use both in the classroom and future physics education research.

ACKNOWLEDGMENTS

The authors would like to thank M. Murdrich for his support regarding the use of Composer in the course. This work was supported by European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie QuSCo grant agreement N^{o} 765267 and by the NSF grant PHY-1734006.

[1] C. Singh, Student understanding of quantum mechanics, American Journal of Physics **69**, 885 (2001), https://doi.org/10.1119/1.1365404.

[2] C. Singh, Student understanding of quantum mechanics at the beginning of graduate instruction, American Journal of Physics **76**, 277 (2008), https://doi.org/10.1119/1.2825387.

[3] G. Zhu and C. Singh, Surveying students' understanding of quantum mechanics in one spatial dimension, American Journal of Physics **80**, 252 (2012), https://doi.org/10.1119/1.3677653.

[4] E. Cataloglu and R. W. Robinett, Testing the development of student conceptual and visualization understanding in quantum mechanics through the undergraduate career, American Journal of Physics **70**, 238 (2002).

[5] G. Zhu and C. Singh, Improving students' understanding of quantum mechanics via the Stern-Gerlach experiment, American Journal of Physics **79**, 499 (2011).

[6] D. A. Zollman, N. S. Rebello, and K. Hogg, Quantum mechanics for everyone: Hands-on activities integrated with technology, American Journal of Physics **70**, 252 (2002).

[7] C. Singh, M. Belloni, and W. Christian, Improving students' understanding of quantum mechanics, Physics Today **59**, 43 (2006).

[8] C. Singh, Student difficulties with quantum mechanics formalism, AIP Conference Proceedings **883**, 185 (2007).

[9] C. Keebaugh, E. Marshman, and C. Singh, Improving student understanding of fine structure corrections to the energy spectrum of the hydrogen atom, American Journal of Physics **87**, 594 (2019).

[10] D. F. Styer, Common misconceptions regarding quantum mechanics, American Journal of Physics **64**, 31 (1996).

[11] R. Müller and H. Wiesner, Teaching quantum mechanics on an introductory level, American Journal of Physics **70**, 200 (2002).

[12] P. J. Emigh, G. Passante, and P. S. Shaffer, Student understanding of time dependence in quantum mechanics, Phys. Rev. ST Phys. Educ. Res. **11**, 020112 (2015).

[13] G. Passante and A. Kohnle, Enhancing student visual understanding of the time evolution of quantum systems, Phys. Rev. Phys. Ed. Res. **15**, 10.1103/PhysRevPhysEducRes.15.01011 (2019).

[14] B. Thaller, *Visual Quantum Mechanics* (Springer-Verlag New York, 2000).

[15] W. Christian, F. Belloni, Mario nd Esquembre, B. A. Mason, L. Barbato, and M. Riggsbee, The Physlet approach to simulation design, The Physics Teacher **53**, 419 (2015).

[16] http://www.falstad.com/mathphysics.html, retrieved 5/30/2020.

[17] http://www.quantum-physics.polytechnique.fr (2019), retrieved 5/30/2020.

[18] L. T. Escalada, N. S. Rebello, and D. Zollman, Student explorations of quantum effects in LEDs and luminescent devices, The Physics Teacher **42**, 10.1119/1.1664385 (2004).

[19] J. R. Hiller, I. D. Johnston, and I. Johnson, *Quantum Mechanics Simulations* (John Wiley and Sons, Inc., 1995).

[20] S. B. McKagan, K. K. Perkins, M. Dubson, C. Malley, S. Reid, R. LeMaster, and C. E. Wieman, Developing and researching PhET simulations for teaching quantum mechanics, American Journal of Physics **76**, 406 (2008), https://doi.org/10.1119/1.2885199.

[21] A. Kohnle, D. Cassettari, T. J. Edwards, C. Ferguson, A. D. Gillies, C. A. Hooley, N. Korolkova, J. Llama, and B. D. Sinclair, A new multimedia resource for teaching quantum mechanics concepts, American Journal of Physics **80**, 148 (2012), https://doi.org/10.1119/1.3657800.

[22] C. Singh, Interactive learning tutorials on quantum mechanics, American Journal of Physics **76**, 400 (2008), https://doi.org/10.1119/1.2837812.

[23] S. B. McKagan, K. K. Perkins, and C. E. Wieman, Deeper look at student learning of quantum mechanics: The case of tunneling, Phys. Rev. ST Phys. Ed. Res. **4**, 10.1103/PhysRevSTPER.4.020103 (2008).

[24] E. Marshman and C. Singh, Investigating and improving student understanding of the expectation values of observables in quantum mechanics, Eur. J. Phys. **38**, 10.1088/1361-6404/aa6d34 (2017).

[25] https://www.quatomic.com/composer/ (2020), retrieved 5/30/2020.

[26] J. J. Sørensen, J. H. M. Jensen, T. Heinzel, and J. F. Sherson, QEngine: A C++ library for quantum optimal control of ultracold atoms, Comp. Phys. Comm. **243**, 135 (2019).

[27] S. Z. Ahmed, J. H. M. Jensen, C. A. Weidner, J. J. Sørensen, and J. F. Sherson, Quantum Composer: Quantum simulation and visualization tool for education and research, arXiv:2006.07263 (2020).

[28] D. J. Griffiths and D. F. Schroeter, *Introduction to Quantum Mechanics, Third Edition* (Cambridge University Press, 2018).