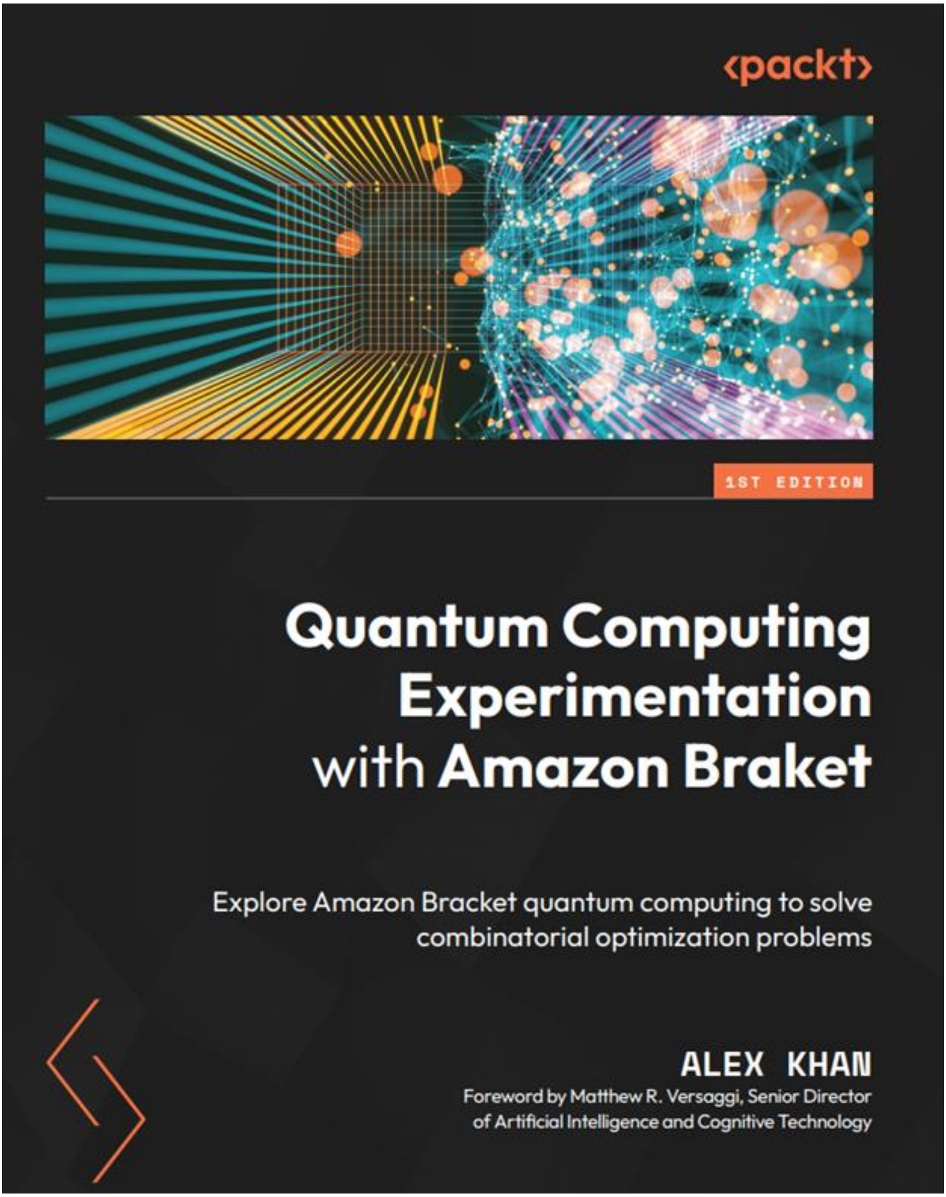

‹packt›

1ST EDITION

Quantum Computing Experimentation with Amazon Braket

Explore Amazon Bracket quantum computing to solve combinatorial optimization problems

ALEX KHAN

Foreword by Matthew R. Versaggi, Senior Director of Artificial Intelligence and Cognitive Technology

https://www.amazon.com/Quantum-Computing-Experimentation-Amazon-Braket/dp/1800565267

Review: “Quantum Computing Experimentation with Amazon Braket” by Alex Khan

by Brian Siegelwax, March 2024

Don’t just learn it. Do it.

Maybe you want to learn quantum computing. Maybe you want your teams to learn quantum computing. Tutorials can be a great start, but you’ll eventually want to dive deeper than even the best tutorials go. Courses can be good, but you have to adhere to their structure. Books are a happy medium, diving deeper than the tutorials but allowing you to learn at your own pace.

A few short years ago, one book was recommended above all: “Quantum Computation and Quantum Information” by Dr. Michael A. Nielsen and Dr. Isaac L. Chuang. Affectionately known as “Mike & Ike” and “The Bible,” it’s still a must-have for any quantum computing library. Along came some copycats, which repackaged the information in Mike & Ike, but leaving Mike & Ike as the one book you couldn’t go wrong with.

Then came “Dancing with Qubits: Find out how quantum computing works and how you can use it to change the world” by Dr. Robert S. Sutor. This book stood out because it was written by a mathematician in language that clearly ought to appeal to readers with mathematical backgrounds. Years later, there are now books written for computer science, engineering, and general backgrounds.

More recently came "Quantum Computing Experimentation with Amazon Braket" by Alex Khan, subtitled "Explore Amazon Braket quantum computing to solve combinatorial optimization problems." This book isn't so much about learning quantum computing, which this book and all introductory books do, but DOING quantum computing. So many courses and books are so focused on theory and mathematics that they neglect to introduce students to actual quantum computers. The whole point of this book is using actual quantum computers.

It's worth noting that quantum computing does not have standardized definitions. Even the term "quantum computing" does not have a standardized definition. Therefore, if you manually solve certain equations with a pencil and paper, many in the field will argue that you're doing quantum computing. Outside of that circle, however, many of us consider "quantum computing" to be the act of using quantum computers, or at least classical simulators of quantum computers. "Quantum Computing Experimentation with Amazon Braket" is focused on using quantum computers and their simulators.

This book will guide you through setting up an Amazon Braket account, accessing the various providers and backends, and running algorithms of commercial interest. That last point is also worth stressing, because most resources focus on the textbook algorithms. These textbook algorithms are important to understanding why we want to use quantum computers and how we might design novel algorithms of commercial interest, but they are not themselves of commercial interest. To the author's credit, he introduces textbook material with real-world applicability in mind.

This book isn't written for a specific background—physics, mathematics, computer science, or engineering—but for Industry. If you want to get yourself or your teams actually using quantum computers and solving sample problems that are legitimately being researched, then "Quantum Computing Experimentation with Amazon Braket" is for you.

For your consideration: The Good, The Bad, and The Ugly.

The Good

Account setup. It's hard to use quantum computers without access to them, so this book walks you through account setup. One of the advantages of using Amazon Braket is that one account gives you access to multiple providers and a variety of quantum computers and simulators.

Code. This isn't the first book with code, but not all books have code. And there isn't just code for algorithms, which you can find elsewhere, but code specifically for getting started with Amazon Braket.

Minimal dependencies. Code with dependencies tends to break. Acknowledging this, the author set out to minimize the potential of external sources to break the code in the book.

Code readability. The author prioritized readability over efficiency. Efficiency is a Python language skill; therefore, the author rightfully prioritized the quantum computing lessons.

Outputs. Sample code outputs are provided to help readers reconcile the results of their experiments. The author explains the probabilistic nature of quantum computers and how variances in the outputs are expected.

Simplifications. Some concepts have been simplified to help readers get started with quantum computing. You can always read more later, but you can also read so much that you never get started. The author provides just enough to get started.

Costs. I don’t believe I’ve seen this in a book before, but the book not only includes pricing for the various backends, but also functions for estimating the costs of your quantum circuits before you submit them.

Visualizations. The book is not just text and mathematics, although it has both. The visualizations are excellent.

Dancing with Qubits. I noted Dr. Sutor’s book at the top of this article, and it’s also in Mr. Khan’s book as a recommendation for further reading.

Plain English. This book is written to be understandable by the masses. That doesn’t mean that all the content is necessarily going to be easy, but it does mean that the average reader shouldn’t need a dictionary to follow along. There will be jargon, sure, but the explanation should read like a casual conversation.

Qiskit. Qiskit is the most popular quantum computing framework, and not everyone knows that you don’t have to use IBM backends with it. If you’re already familiar with Qiskit and don’t want to learn the Amazon Braket SDK, you don’t have to. You can continue to use Qiskit, but you can save money by using Amazon Braket backends.

Comparing solvers. This is just good practice. A solver can look like its working with a toy problem, and then we prematurely celebrate that. If we compare multiple solvers—classical, hybrid, and quantum—we may discover that a solver that works is not necessarily the best solver.

IBM alternatives. For reasons beyond the scope of this review, IBM Quantum needs alternatives. This book presents Rigetti and OQC as two such alternatives.

Personalized copy. I’ve got a personalized copy of this book, and I believe it’s my first ever personalized book. I can’t make any promises, but maybe if you reach out to Mr. Khan you can send him your book and he can autograph it for you. Or try to link up with him at a conference and bring your copy with you.

The Bad

Too soon. The book mentions QuEra’s “Aquila,” but it was published a month before it became available. Fortunately, the book is still useful for setting up your Amazon Braket account to access “Aquila,” a 256-qubit neutral atom quantum computer, and the largest publicly available quantum computer in the world at the time I’m writing this. Alternately, the Amazon Braket Python SDK code in the book may also be useful for accessing “Aquila” through the qBraid platform, where you can get 1,000 free credits to try it out.

Breaking changes. They’re unavoidable. The quantum industry needs a whole lot more software architects to avoid this in the future. In the meantime, the author has tried his best to keep up with these changes as they happen. In principle, the Amazon Braket Python SDK should break less frequently than Qiskit, which actually schedules breaking changes.

Quantum annealing. Although you can solve toy problems with them, there are no theoretical or experimental computational advantages to using D-

Wave annealers. However, since the book's publication, NEC and Qilimanjaro have announced initiatives to improve quantum annealing with fewer-but-higher-quality qubits. It remains to be seen if there will be any advantages to using these devices, or if they will be listed through Amazon Braket, but it seems plausible that the information on quantum annealing in the book ought to be applicable to some extent if they prove useful.

QAOA. Although I don't personally subscribe to QAOA or any other hybrid classical-quantum algorithms, also due to the absence of advantages to do so, the fact of the matter is that they're popular. And if you absolutely insist on using QAOA, premium IBM Quantum access will run you $1.60 per second. If you're using the same generic algorithm, using Rigetti through Amazon Braket should save you considerable money. A question mark gets raised when using Q-CTRL's Fire Opal with IBM Quantum because their QAOA solver will greatly outperform a generic algorithm, so that would be worthy of investigation.

IonQ. I would personally prefer to use Quantinuum, AQT, or QSCOUT ion trap quantum computers. If you neither know nor care my reasons why, IonQ is admittedly easier to access. Quantinuum is the industry leader in a performance metric called Quantum Volume (QV), but the pricing through Azure Quantum is unintuitive, to put it nicely. QSCOUT access requires a proposal and AQT access requires contacting the company. So, if you want to explore ion trap quantum computing, Mr. Khan can show you the easiest path to doing so.

Comparing solvers. Although I also listed this under “The Good,” comparisons may have different results at different scales. A solver that looks like the “winner” with a toy problem, might break quickly when scaling the problem up. It’s important to not immediately stop performing comparisons after an initial result.

The Ugly

Delisting. Since publication, D-Wave and Xanadu are no longer accessible through Amazon Braket. In fact, Borealis is offline altogether. It honestly surprised me when they were listed, actually, because D-Wave has been offering limited free usage for years through their cloud, and Xanadu offered Borealis for free through their cloud. You could argue that D-Wave’s inclusion allowed access beyond the free monthly allotment, but I’ve spoken to people who didn’t know they could’ve tried it for free.

Conclusion

“Quantum Computing Experimentation with Amazon Braket” is a well-written, well-illustrated book, and I appreciate its emphasis on using quantum computers and on problems of commercial interest. Although I expressed personal disinterest in those problems, that’s not the point. The solvers that we are most interested in can’t be run on the quantum computers of today; the hardware isn’t mature enough yet. But the point of having these other solvers in the book is twofold: 1) to deep dive what is being tried today, and to then 2) encourage exploration of novel advantageous solvers. I applaud that.

The book’s first strength would be accessing QuEra’s “Aquila,” even though there isn’t yet a chapter for using “Aquila.” “Aquila” is available primarily though Amazon Braket, and it’s the quantum computer most worth watching at the moment. This has to do with QuEra’s roadmap, which is beyond the scope of this review.

The book’s second strength would be using Rigetti, OQC, and the Amazon Braket simulators as alternatives to IBM Quantum, its quantum computers, and its simulators. Between the partner fees, the $1.60 per second for premium access, and the scheduled breaking changes, plausible alternatives are desperately needed.

It’s worth noting how the book highlights the rapid shifting in the field of quantum computing. In the short time since publication, code has changed, some of the quantum computers have changed, and even some of the providers have changed. Hats off to Mr. Khan for keeping up with those changes while this book was still in development.

His effort paid off, because this book is worthy of inclusion in your library.