Implementation Guide

Quantum Computing Exploration for Drug Discovery on AWS

Quantum Computing Exploration for Drug Discovery on AWS: Implementation Guide

Table of Contents

An open-source solution for studying drug discovery problems using classical and quantum compute resources

Publication date: *May 2022 (last update: June 2023)*

Quantum Computing Exploration for Drug Discovery on AWS (QCEDD) helps you study drug discovery problems using quantum computing (Amazon Braket), such as molecular docking and protein folding. You can use Amazon Braket to call quantum computing resources for experiments and support user-customized algorithms to adapt to research findings in more scenarios.

This solution includes the following features:

- One-click deployment of the Jupyter environment required for quantum computing algorithms.
- A fully managed Jupyter Notebook, which can be used to design and debug drug discovery algorithms.
- Ability to customize algorithms for other drug discovery problems and reuse them.

This implementation guide includes a workshop with a series of notebook experimentation. For example, the workshop introduces the molecular unfolding algorithm from Mato, Kevin, et al. to illustrate the background information, how to build a model, optimize its configuration, and batch evaluate the experiment results.

This implementation guide describes architectural considerations and configuration steps for deploying Quantum Computing Exploration for Drug Discovery on AWS in the Amazon Web Services (AWS) Cloud. It includes links to an AWS CloudFormation template that launches and configures the AWS services required to deploy this solution using AWS best practices for security and availability.

The guide is intended for researchers, data scientists, algorithm engineers in the drug discovery field and quantum computing advocates with architecture hands-on experience in the AWS Cloud.

Cost

You are responsible for the cost of AWS services used when running this solution. The actual cost depends on the tasks run and their complexity. As of this revision, the cost factors mainly consist of five types:

- Amazon Sagemaker Notebook
- Amazon S3
- Amazon Elastic Container Registry
- Amazon Braket Hybrid Jobs

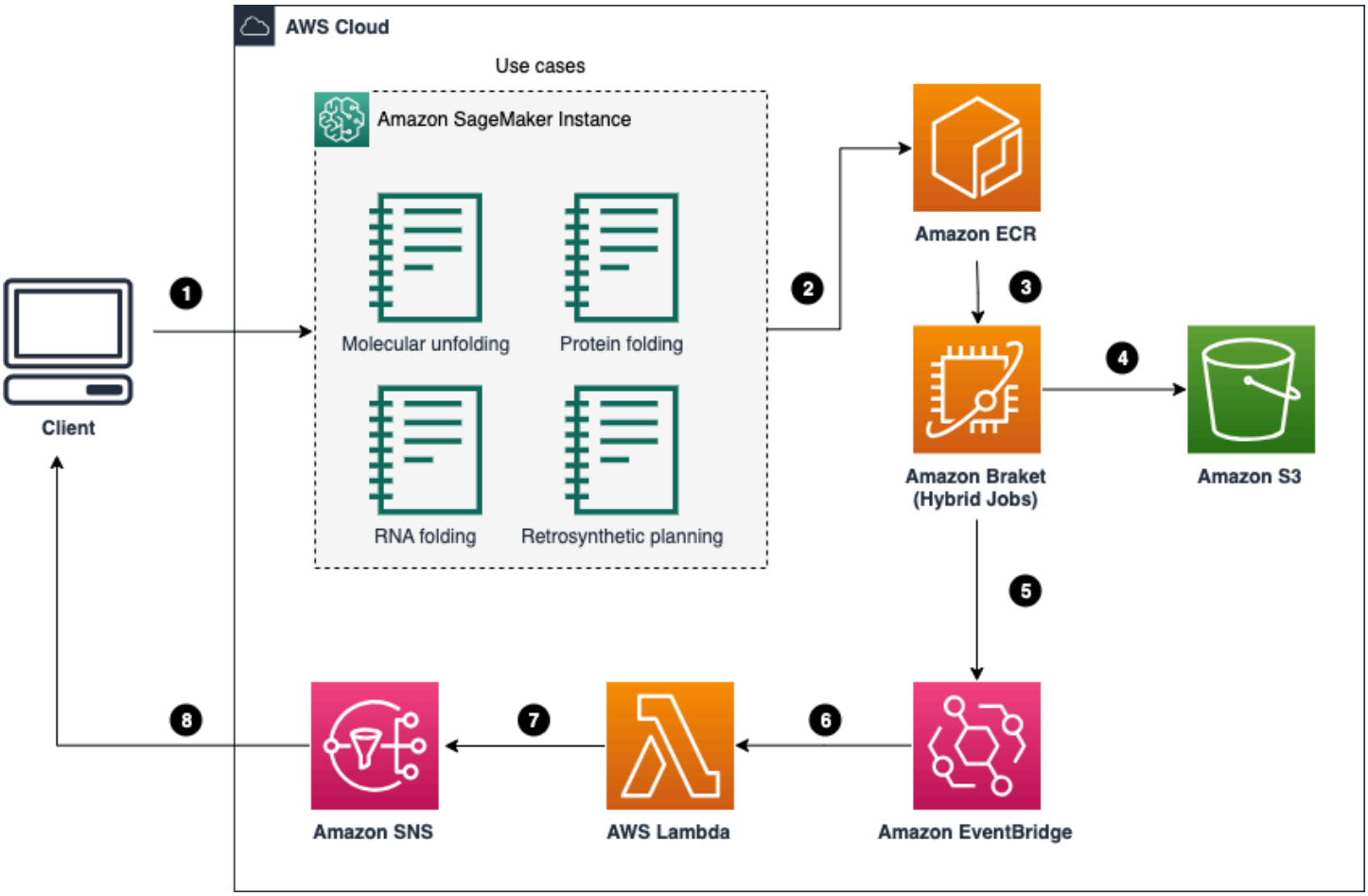
If the customer only uses the notebook to explore the algorithm, the cost factors mainly involve the notebook and the computing from Amazon Braket task. If the customer evaluates different use cases with different parameters, the cost factors mainly involve the computing from Amazon Braket Hybrid Jobs.

AWS service	Resource	Dimensions	Cost
Amazon Sagemaker Notebook	ml.c5.xlarge	long run instance	$4.90 per day
Amazon S3		<1G	$0.02
Amazon Elastic Container Registry	images for different use cases	<1G	$0.02
Amazon Braket Hybrid Jobs	ml.m5.large runs and ml.m5.4xlarge (Molecular unfolding)	one complete experiment consists of 12 jobs for different parameters, ml.m5.large runs for 44 minutes and	$0.76 per complete experiment

AWS service	Resource	Dimensions	Cost
		ml.m5.4xlarge runs for 44 minutes in total	
	ml.m5.large runs and ml.m5.4xlarge (RNA unfolding)	one complete experiment consists of 2 jobs for different parameters, ml.m5.large runs for 450 minutes and ml.m5.4xlarge runs for 429 minutes in total	$7.46 per complete experiment
Total Cost			$13.16 per day

Architecture overview

Deploying the Quantum Computing Exploration for Drug Discovery on AWS solution with the default parameters builds the following environment in the AWS Cloud.



Quantum Computing Exploration for Drug Discovery on AWS architecture

1. This solution deploys a notebook instance to allow Amazon SageMaker Notebook users to conduct **notebook experiments**. Notebooks come with use cases for different drug discovery problems such as molecule unfolding, RNA folding, and protein folding.
2. When used for the first time, the system will mirror and upload the dependent packages required for the experiment to Amazon ECR.
3. This program uses Amazon Braket Hybrid Job for experiments.
4. The experiment results are stored in Amazon S3.
5. Run multiple Hybrid Jobs in one experiment, and initiate the events in Amazon EventBridge when the Hybrid Job is completed.

6. Events initiated by Amazon EventBridge are sent to Lambda for analysis and summary.
7. Lambda sends an analysis report to Amazon SNS, and all subscribers who have subscribed to this topic will receive this report by email. This step is optional, and you can specify emails for subscription notifications when deploying the solution.
8. Return to SageMaker Notebook to run the code to analyze and display the experiment results.

Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This shared responsibility model reduces your operational burden because AWS operates, manages, and controls the components including the host operating system, the virtualization layer, and the physical security of the facilities in which the services operate. For more information about AWS security, visit the AWS Cloud Security.

Security groups

The security groups created in this solution are designed to control and isolate network traffic between the solution components. We recommend that you review the security groups and further restrict access as needed once the deployment is up and running.

Amazon Braket security

Refer to Security in Amazon Braket to learn how to apply the shared responsibility model when using Amazon Braket.

Consider using Amazon Macie with Amazon S3

Amazon Macie is a data security and data privacy service that uses machine learning and pattern matching to help you discover, monitor, and protect sensitive data in your AWS environment. Macie automates the discovery of sensitive data, such as personally identifiable information (PII) and financial data, to provide you with a better understanding of the data that your organization stores in Amazon S3.

Macie also provides you with an inventory of your S3 buckets, and it automatically evaluates and monitors those buckets for security and access control. If Macie detects sensitive data or potential issues with the security or privacy of your data, it creates detailed findings for you to review and remediate as necessary. For more information, refer to the *Amazon Macie User Guide*.

Design considerations

Regional deployments

This solution uses some services which may not be currently available in all AWS Regions. Launch this solution in an AWS Region where required services are available. For the most current availability by Region, refer to the AWS Regional Services List.

As of this revision, this solution is supported in the following AWS Regions:

Region ID	Region name
us-east-1	US East (N. Virginia)
us-west-1	US West (N. California)
us-west-2	US West (Oregon)
eu-west-2	Europe (London)

AWS CloudFormation template

This solution uses AWS CloudFormation to automate the deployment of the Quantum Computing Exploration for Drug Discovery on AWS with all associated components in the AWS Cloud. It includes the following CloudFormation template, which you can download before deployment:

View template

quantum-computing-exploration-for-drug-discovery.template: Use this template to launch the solution and all associated components. The default configuration deploys Amazon SageMaker Notebook, Amazon Braket, AWS Identity and Access Management (IAM), Amazon EventBridge, and Amazon SNS, but you can customize the template to meet your specific needs.

Automated deployment

Before you launch the solution, review the architecture, supported Regions, and other considerations discussed in this guide. Follow the step-by-step instructions in this section to configure and deploy the solution into your account.

Time to deploy: Approximately 15 minutes.

Note: Before deploying this solution, we recommend you create a billing alarm to monitor your estimated AWS charges.

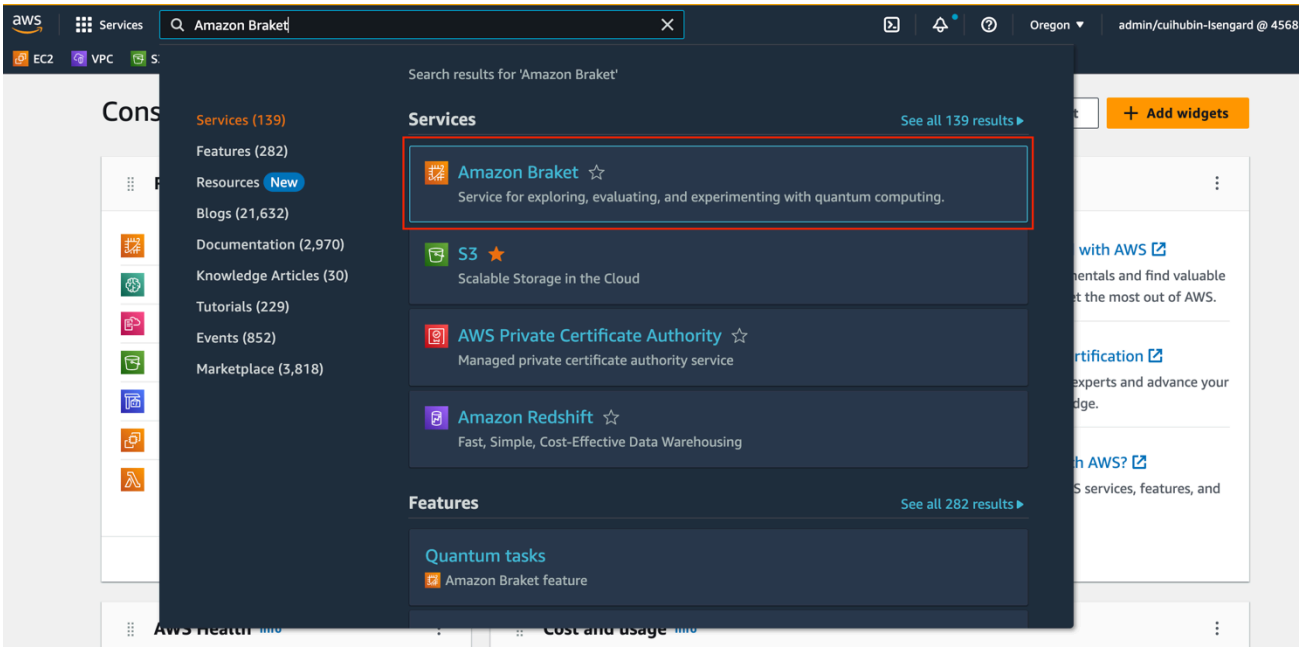
The user or IAM role to perform the deployment must have at least permissions defined in the permissions.json file. If you use the permissions defined in this file to deploy this solution, your CloudFormation stack name should start with `QCEDD`, for example, `QCEDDStack`.

Prerequisites

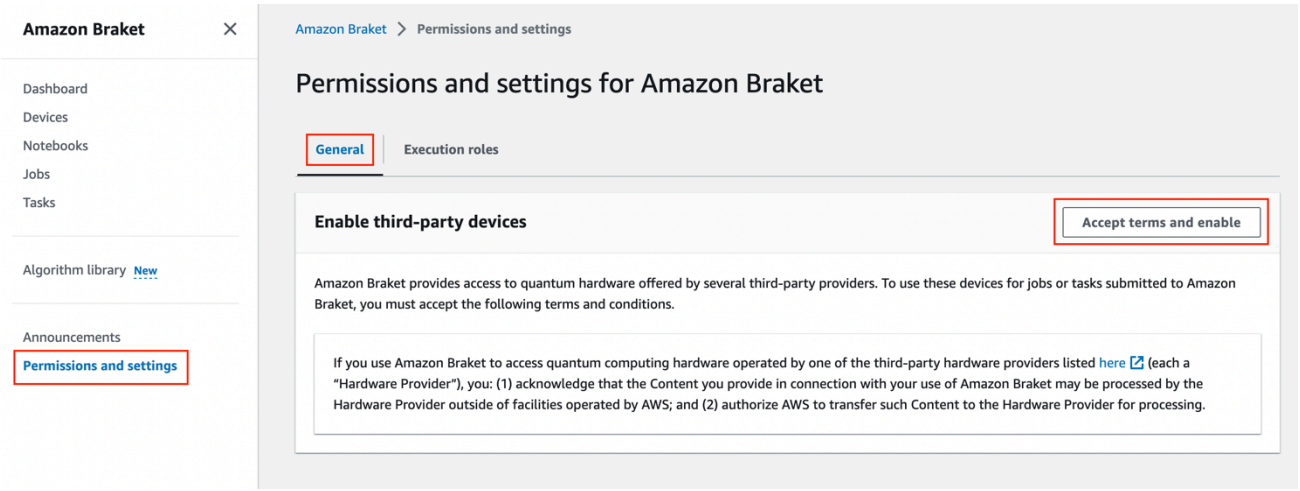
If bellow steps have already been done, please skip this section.

Activate Amazon Braket service

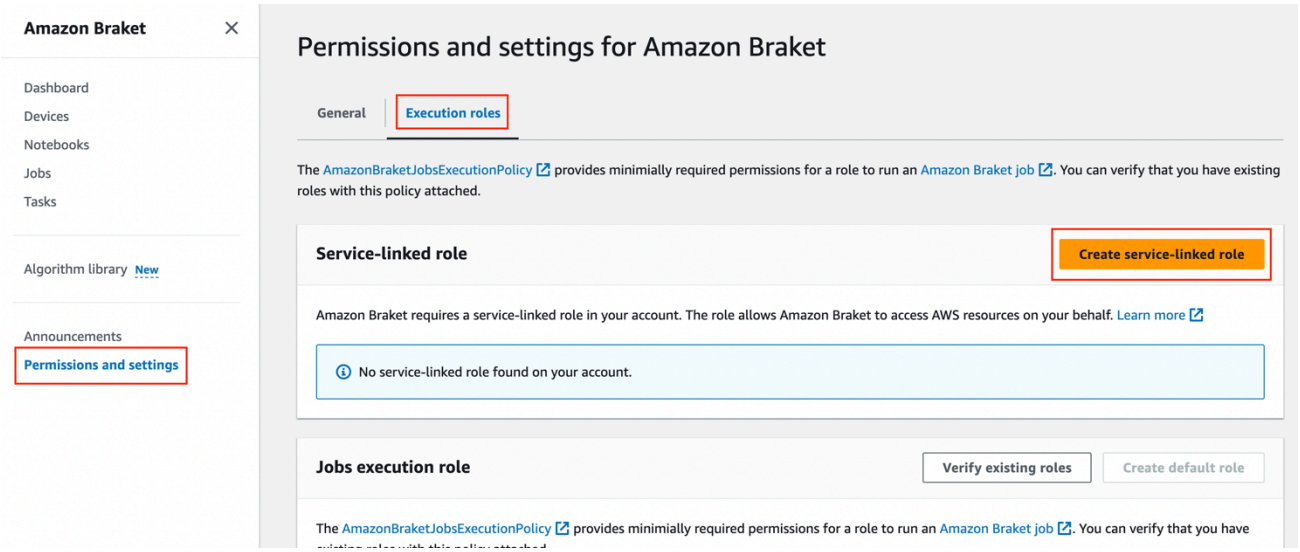
1. Sign in to the AWS management console, and search for Amazon Braket.



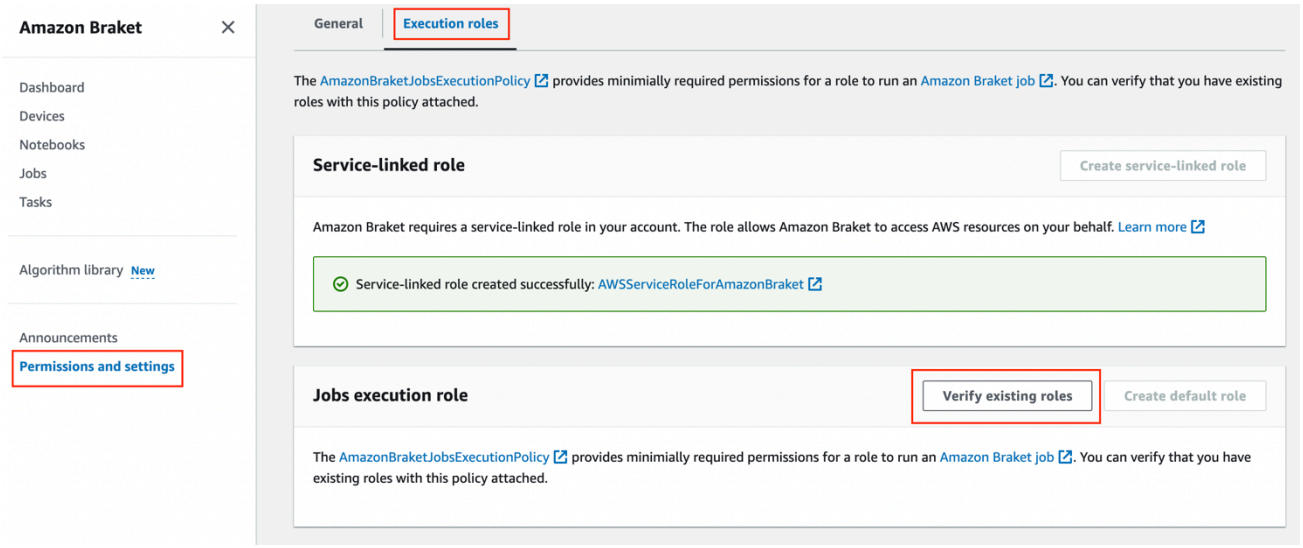
2. Choose **Permissions and Settings** in the left navigation pane, then choose the **General** tab, and choose **Accept and Enable**.



3. Choose the **Execution Roles** tab, and choose **Create Service-linked role** to create the service linked role.



4. Choose **Verify Existing Roles** to create jobs execution role.



Step 1: Launch the AWS CloudFormation template into your AWS account

1. Sign in to the AWS Management Console and select the button to launch the AWS CloudFormation template.

   Launch solution

   Alternatively, you can download the template as a starting point for your own implementation.
2. The template launches in the US West (Oregon) by default. To launch this solution in a different AWS Region, use the Region selector in the console navigation bar.
3. On the **Create stack** page, verify that the correct template URL is in the **Amazon S3 URL** text box and choose **Next**.
4. On the Specify stack details page, assign a name to your solution stack. For information about naming character limitations, refer to IAM and AWS STS quotas in the *AWS Identity and Access Management User Guide*.
5. Under **Parameters**, review the parameters for the template and modify them as necessary.

   This solution uses the following parameters.

Parameter	Default	Description
snsEmail	null	Email to subscribe messages.

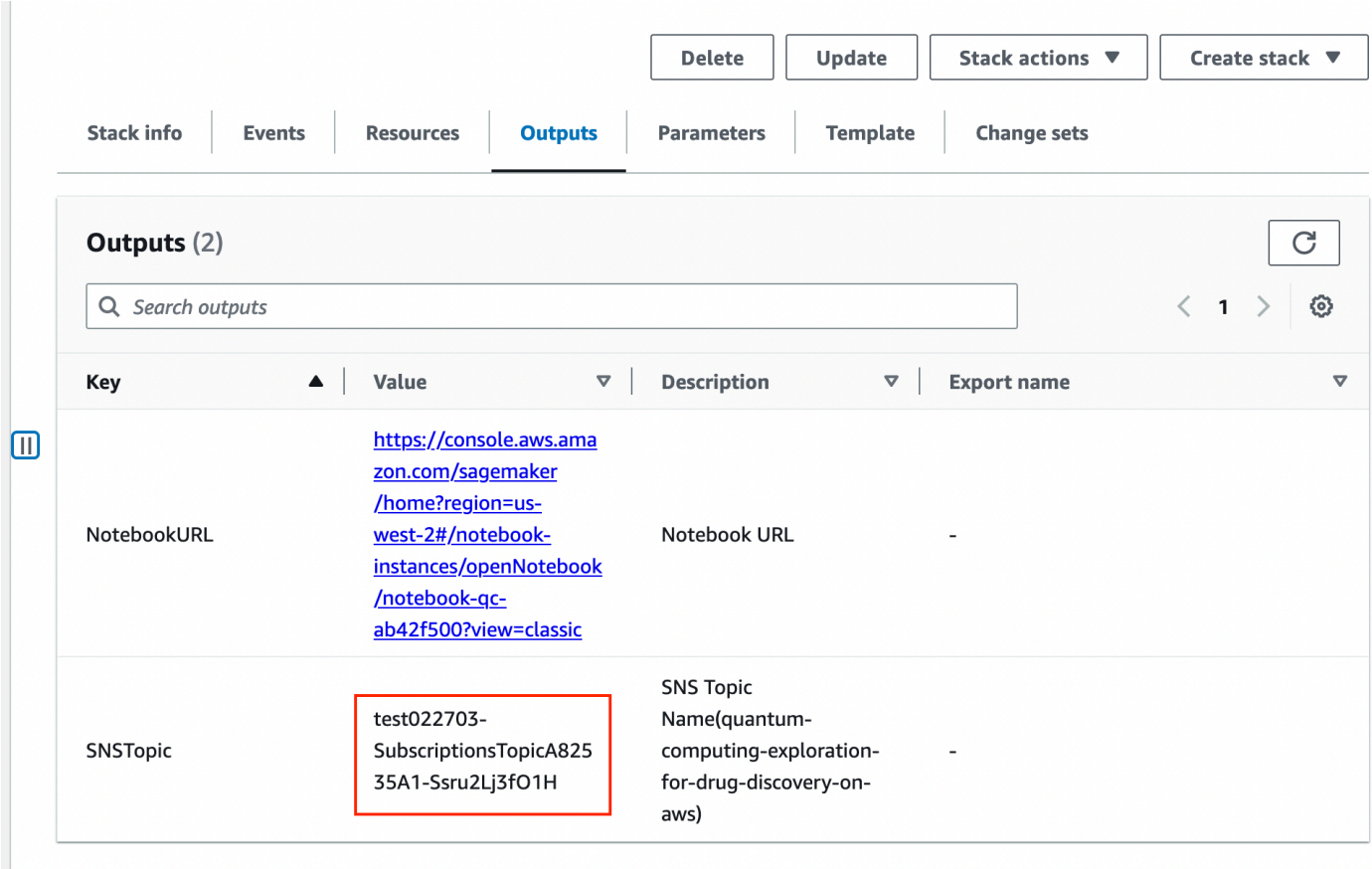
6. Choose **Next**.
7. On the **Configure stack options** page, choose **Next**.
8. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.
9. Choose **Create stack** to deploy the stack.
10. If you configured a subscription email, please pay attention to the email inbox, and then click the **Confirm Subscription** link to confirm the subscription.

You can view the status of the stack in the AWS CloudFormation Console in the **Status** column. You should receive a **CREATE_COMPLETE** status in approximately 15 minutes.

Step 2: (Optional) Subscribe to SNS notification

Follow below steps to subscribe to SNS notification via email to receive result notifications from AWS Step Functions. You can also subscribe to the notification via text messages.

1. Sign in to the AWS CloudFormation console.
2. On the **Stacks** page, select the solution's root stack.
3. Choose the **Outputs** tab and record the value for the SNS topic.



SNS topic value on the solution stack's Outputs tab

4. Navigate to the Amazon SNS console.
5. Chose **Topics**, then select the SNS topic that you obtained from the CloudFormation deployment output.
6. Choose **Create subscription**.
7. Select **Email** from the **Protocol** list.
8. Enter your email in **Endpoint**.
9. Choose **Create subscription**.
10. Check your inbox for the email, and select the **Confirm Subscription** link to confirm the subscription.

Update AWS CloudFormation stack

If needed, you can update the AWS CloudFormation stack to change the deployment settings.

Important

If the content of the notebook algorithm changed, please delete the original CloudFormation and create a new CloudFormation. Updating the CloudFormation will not make your algorithm changes take effect.

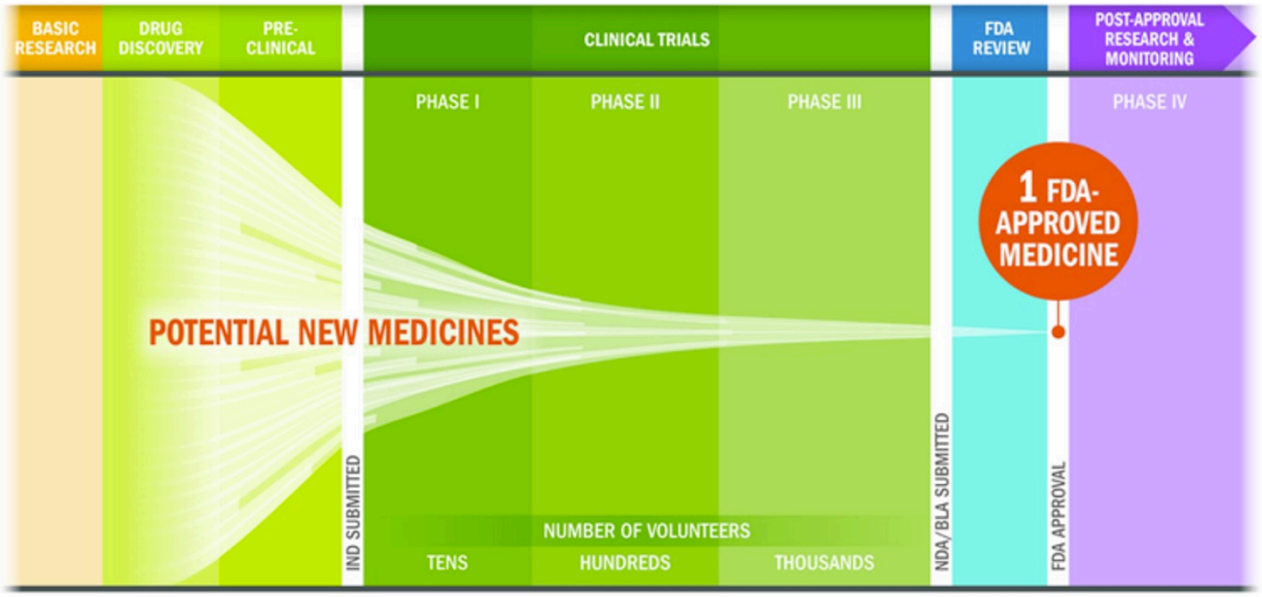
1. Sign in to the AWS CloudFormation console.
2. Select the root stack of this solution, not the nested stack.
3. Choose **Update**.
4. Select **Use current template**, and choose **Next**.
5. Update the parameters as needed, and choose **Next**.
6. On the **Configure stack options** page, choose **Next**.
7. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.
8. Choose **Update stack** to update the stack.

Workshop

Background

As of this revision, quantum computing technology is in an early phase, and its full impacts are still being understood (Quantum computing use cases). We are still in the noisy intermediate-scale quantum (NISQ) era, where the leading quantum processors contain 50 to a few hundred qubits, but are not advanced enough to reach fault-tolerant stage. When we are one day in fault-tolerant quantum computing (FTQC), we can run really powerful algorithms in quantum computers.

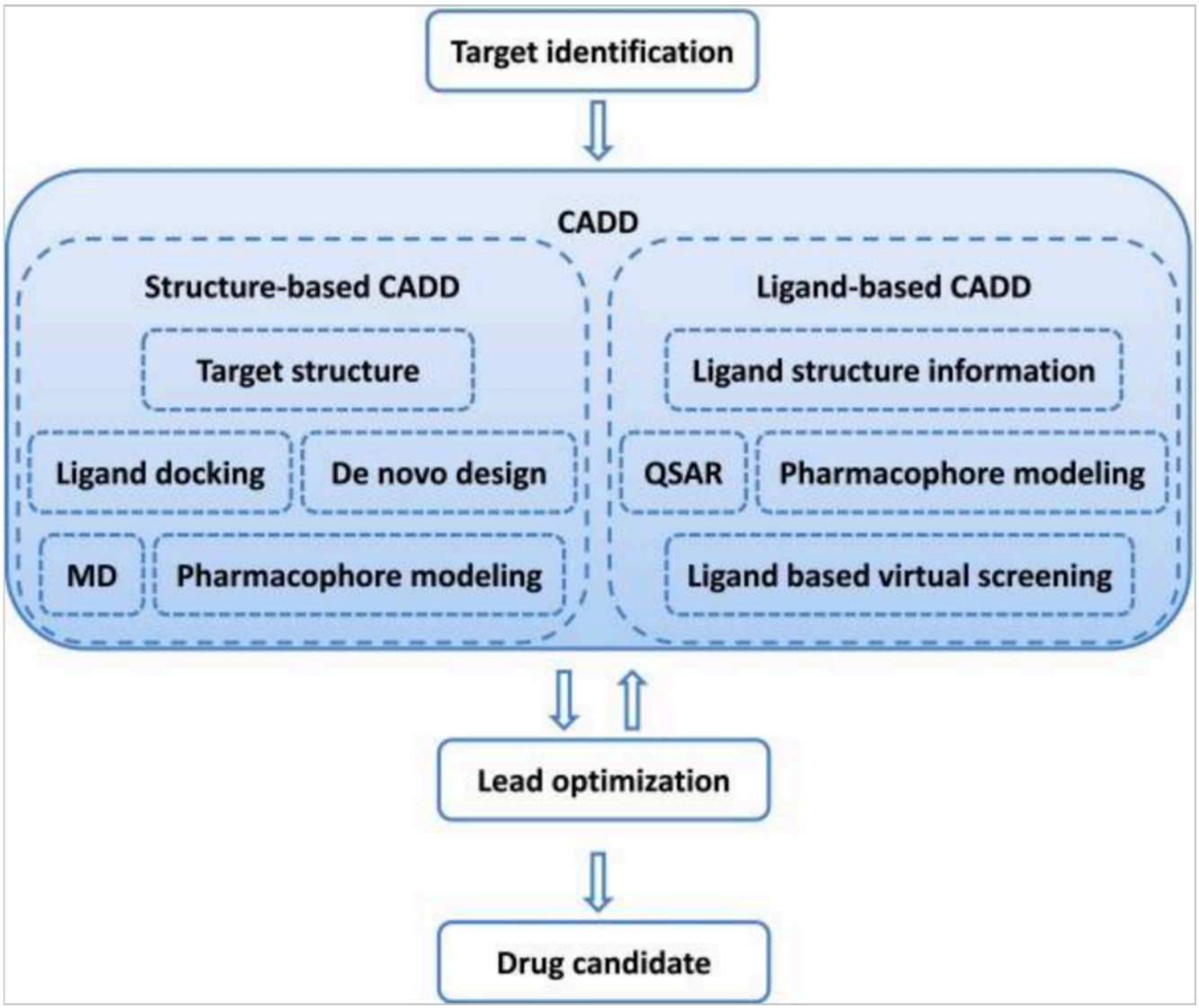
Drug discovery is one area where computational researchers are aiming to assess the state of quantum computing. Drug discovery is costly and time-intensive process with high rate of failure of drug candidates. The average R&D cost required to bring a new, FDA approved medicine to patients is estimated to be $2.6 billion. This includes the cost of many potential medicines that do not make it through to FDA approval. For example, in research for treatments for Alzheimer's disease over the past 16 years, only four new medicines have been approved out of 123 treatment attempts that were tested in clinical studies. That's a three percent approval rate.



Drug Discovery

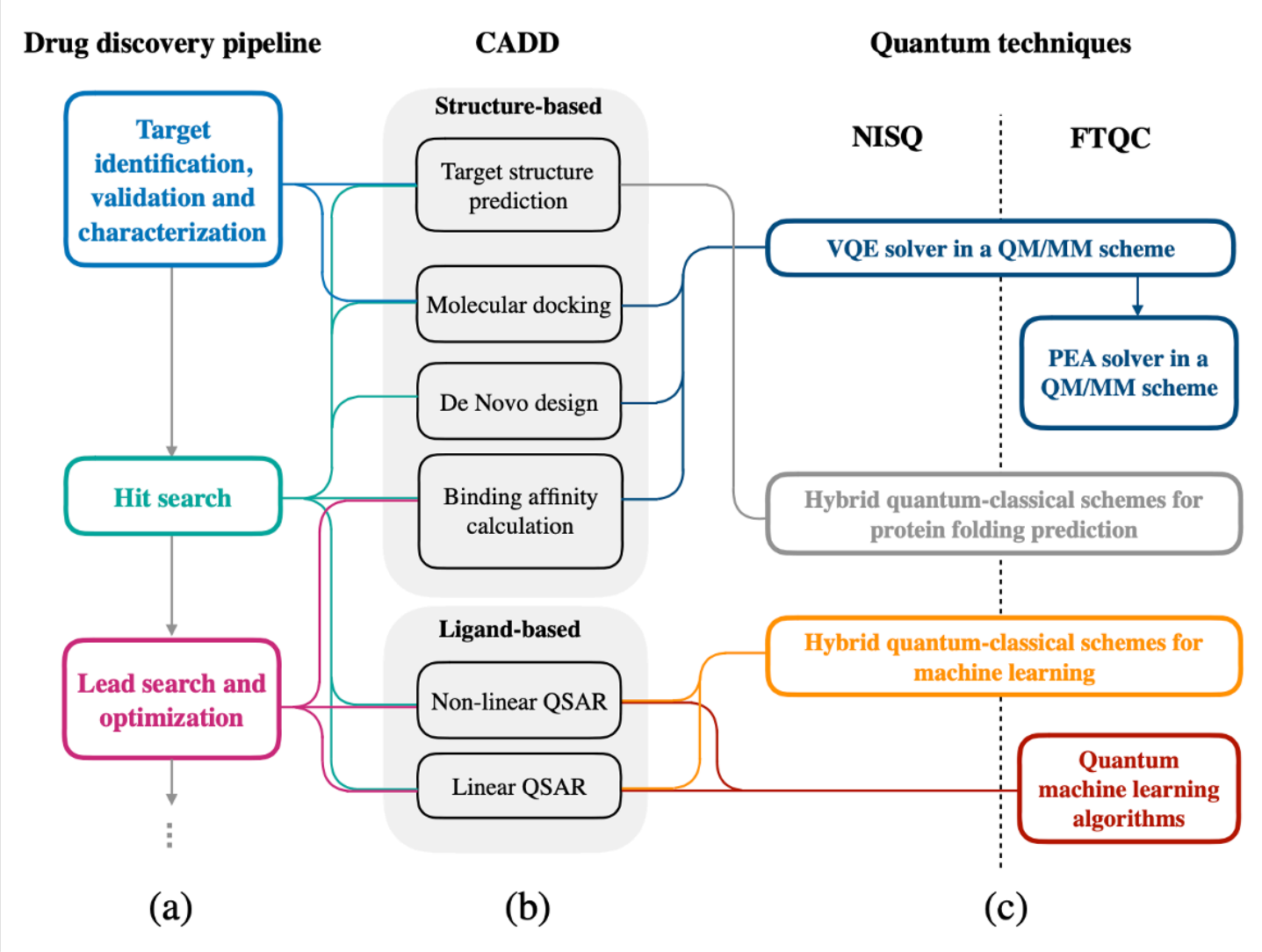
To tackle these problems, innovation and advancements in computer-aided drug design (CADD) aim to increase the productivity in drug research and development. The following figure (CADD) illustrates where CADD approaches, including structured-based methods and ligand-based

methods, can fit within the drug discovery process. In particular, drug discovery researchers have identified quantum computing techniques that may be applicable to the components in the general workflow of drug discovery process (CADD-QC).



CADD

While not as advanced as Fault-tolerant quantum computing (FTQC) shown in the figure CADD-QC, Amazon Braket already provides access to Noisy intermediate-scale quantum computing (NISQ). Therefore, it might be beneficial for customers at the forefront of innovations in their fields to start with quantum computing to research application, to build expertise, and secure Intellectual Property (IP).



CADD-QC

References

1. “Cost of Developing a New Drug”. Tufts CSDD & School of Medicine and US FDA Infographics, Nov.2014
2. The drug development and approval process is about much more than the final “okay”
3. Sliwoski, Gregory, et al. "Computational methods in drug discovery." Pharmacological reviews 66.1 (2014): 334-395
4. Cao, Yudong, Jhonathan Romero, and Alán Aspuru-Guzik. "Potential of quantum computing for drug discovery." IBM Journal of Research and Development 62.6 (2018): 6-1.

Molecular Unfolding – Quadratic Unconstrained Binary Optimization

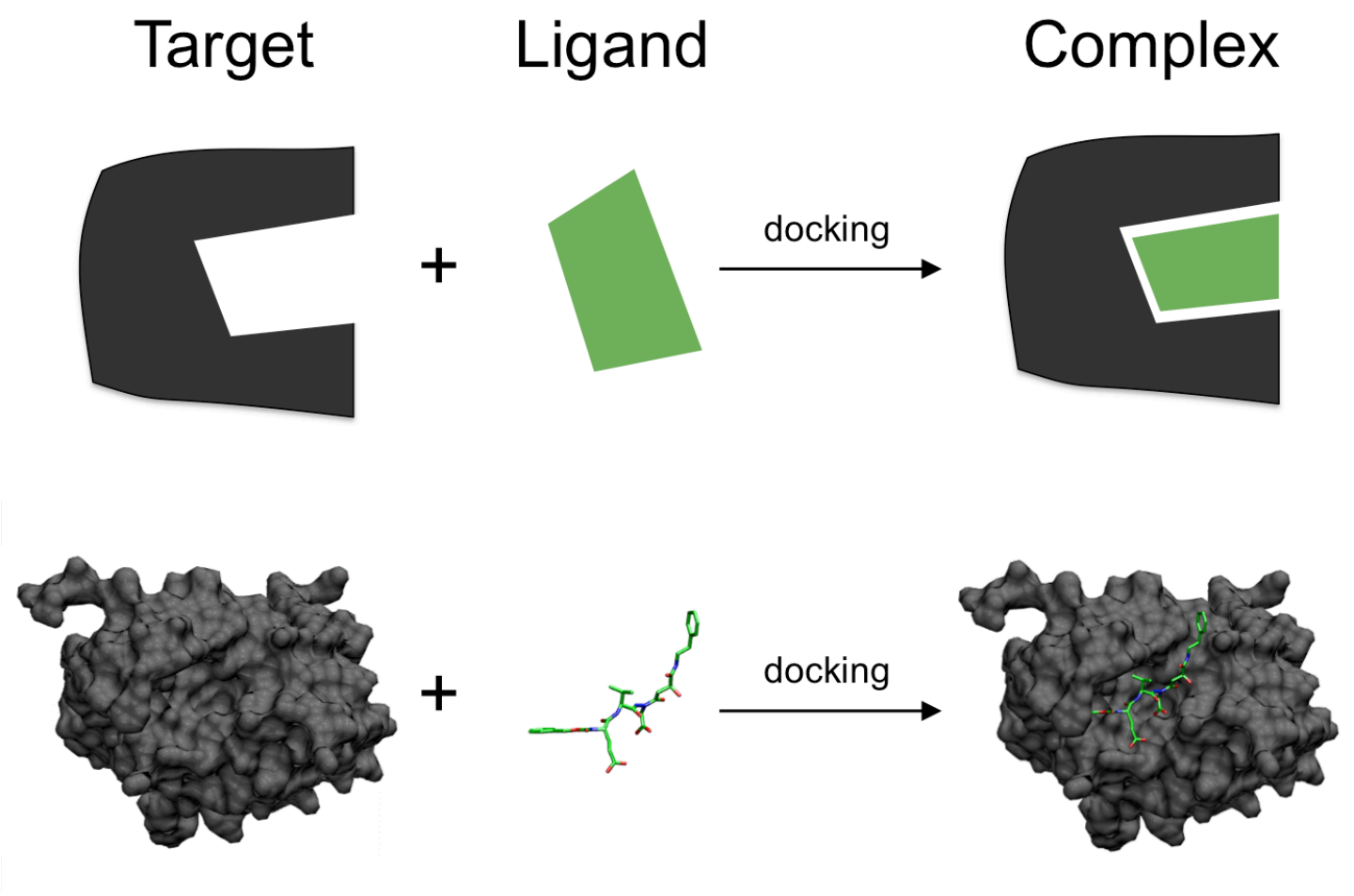
Notebook experimentation

The workshop implements the work from Quantum Molecular Unfolding and the Molecular Unfolding with Quantum Annealing video in Amazon Braket.

Molecular docking

Molecular docking (MD) is an important step of the drug discovery process, which aims at calculating the preferred position and shape of one molecule to a second when they are bound to each other. This step focuses on computationally simulating the molecular recognition process. It aims to achieve an optimized conformation for both the protein and ligand and relative orientation between protein and ligand such that the free energy of the overall system is minimized.

In this work, the protein or the pocket is considered as a rigid structure, and the ligand is considered as a flexible set of atoms.



Molecular docking

As described in Quantum Molecular Unfolding, published by Mato et al, there are usually three main phases in MD:

1. Expansion of the ligand to an unfolded shape, to reduce bias, that is, molecular unfolding (MU). MU includes:
   - Identification of the rotatable bonds
   - Internal distance maximization
   - Removal of tool related bias (for example, SMILES-to-3D)
2. Initial placement, which includes:
   - Ligand main fragments decomposition
   - Ligand initial poses identification
   - Placement of the ligand into the pocket with rigid roto-translations
3. Shape refinement, which includes:

- Use of the rotatable bonds to modify the ligand shape and to match the protein pocket
- Docking score maximization

Here we focus on the first phase of MD, ligand expansion, which aims to reduce shape bias that may affect the final quality of docking. In this solution we use the quantum annealing approach to molecular unfolding (MU) as published by Mato et al.

After deployment, you can go to select the solution's root stack on the **Stacks** page, choose the **Outputs** tab, and open the link for your notebook. See the file **healthcare-and-life-sciences/a-1-molecular-unfolding-quadratic-unconstrained-binary-optimization/molecular-unfolding-qubo.ipynb** for more details.

References:

Wiki: Molecular Docking

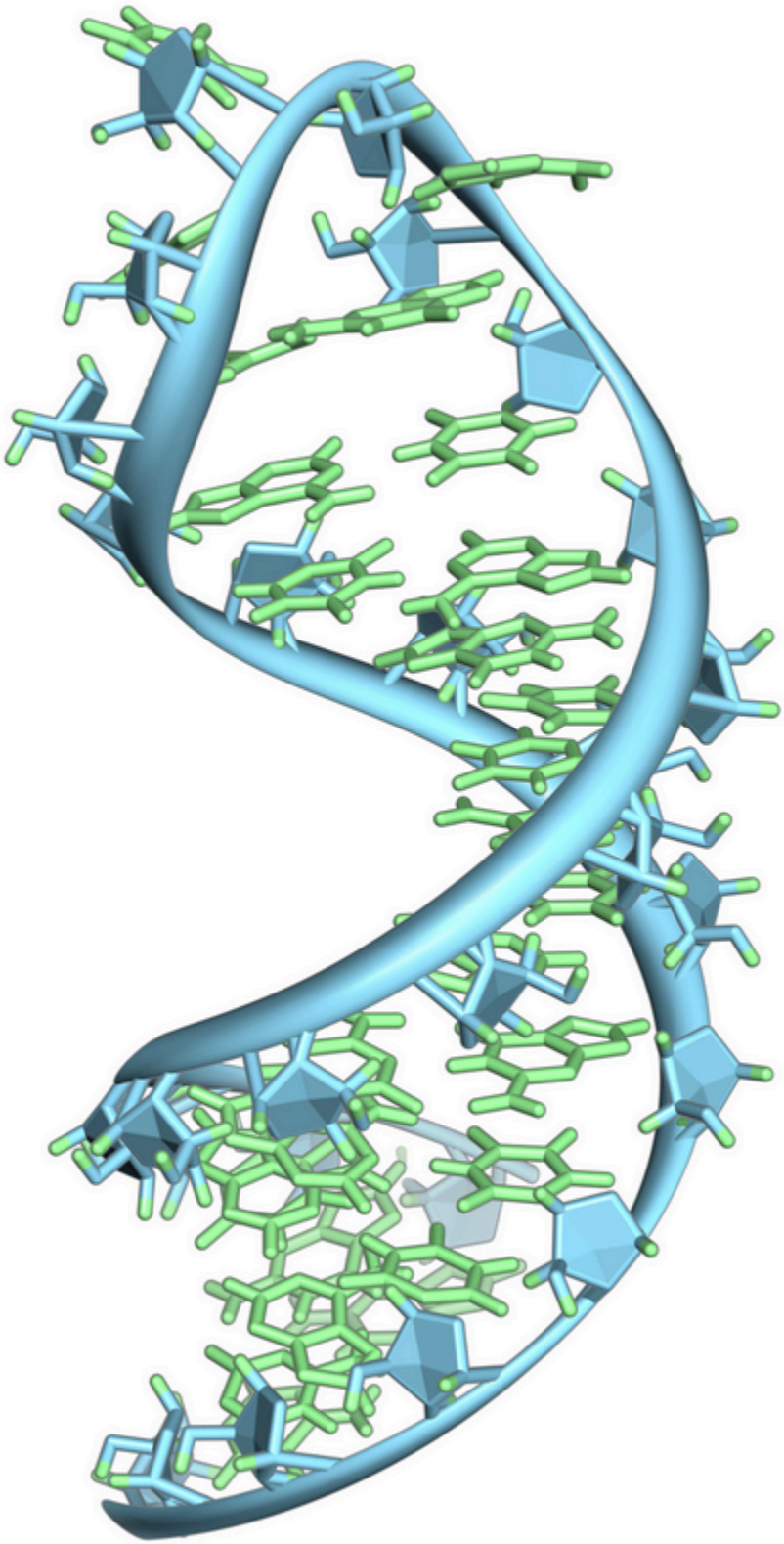
RNA Folding – Quadratic Unconstrained Binary Optimization

Notebook experimentation

This notebook implements the work from RNA Folding Using Quantum Computers inspired by the work from QHack 2022 Winner.

RNA folding

RNA folding refers to the process by which a single-stranded RNA molecule adopts a specific three-dimensional structure through the formation of intramolecular base pairs. This process is crucial for the proper functioning of RNA in a variety of biological processes, such as gene expression, RNA splicing, and protein synthesis.



A hairpin loop from a pre-mRNA

In this work, the quantum annealer is leveraged to predict the secondary structure of RNA. To steer the system towards maximizing both the number of base pairs and the average length of the stems, a Hamiltonian is formulated in the Binary Quadratic Model (BQM) format.

After deployment, you can go to select the solution's root stack on the **Stacks** page, choose the **Outputs** tab, and open the link for your notebook. See the file **healthcare-and-life-sciences/b-1-rna-folding-quadratic-unconstrained-binary-optimization/rna-folding-qubo.ipynb** for more details.

References:

Wiki: RNA

Protein Folding – Quantum Random Walk

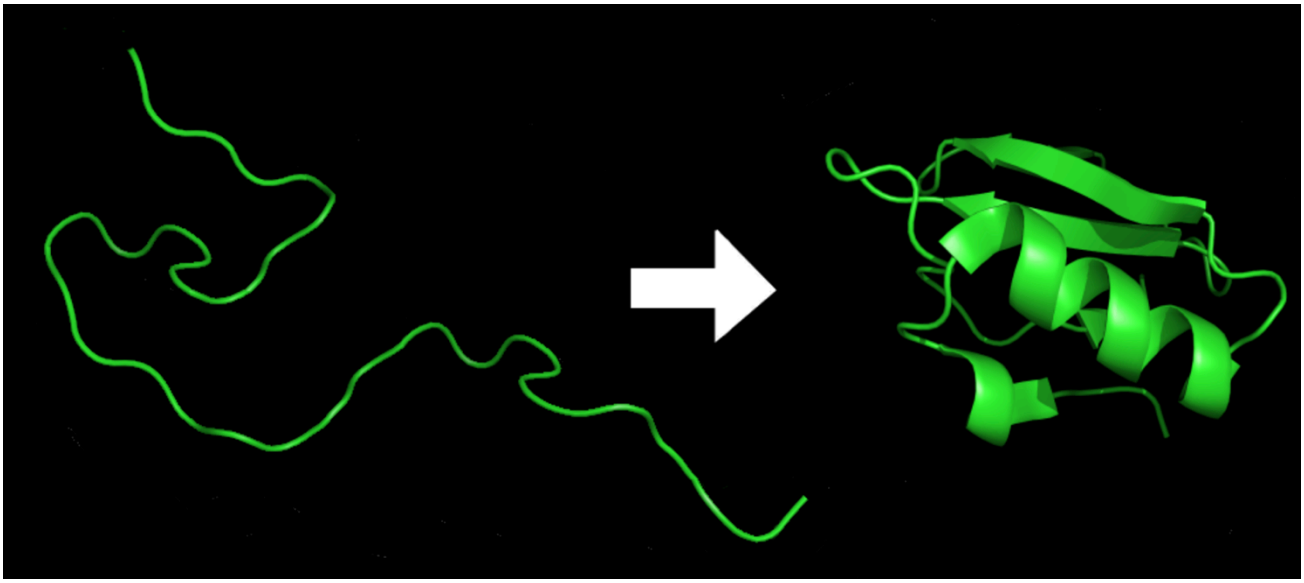
Notebook experimentation

This notebook implements the QFold: quantum walk and deep learning to solve protein folding in Amazon Braket. This is mainly contributed by Roberto Campos based on his implementation.

Protein folding using quantum random walk

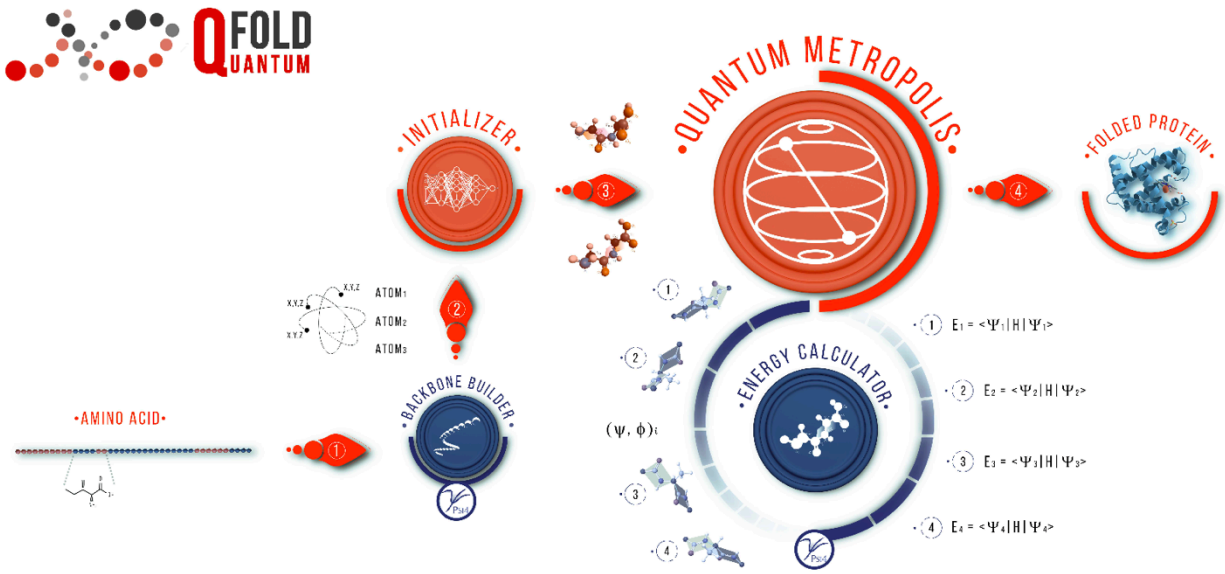
Protein folding is the process by which a protein molecule assumes its three-dimensional shape, which is essential for its proper function. Proteins are made up of a linear chain of amino acids, and their final structure is determined by the sequence of amino acids and the interactions between them.

During protein folding, the linear chain of amino acids folds into a unique three-dimensional structure, which is stabilized by various types of interactions, such as hydrogen bonds, electrostatic forces, and van der Waals forces. The process of protein folding is highly complex and involves multiple stages, including the formation of secondary structures, such as alpha-helices and beta-sheets, and the packing of these structures into a final three-dimensional shape.



Protein before and after folding

In this work, the quantum walks are applied to a Metropolis algorithm in order to predict how proteins fold in 3D. Quantum walks are quantum analogues of classical random walks. In contrast to the classical random walk, where the walker occupies definite states and the randomness arises due to stochastic transitions between states, in quantum walks randomness arises through: (1) quantum superposition of states, (2) non-random, reversible unitary evolution and (3) collapse of the wave function due to state measurements. This is named as Qfold in the original paper.



Scheme of the QFold algorithm

After deployment, you can go to select the solution's root stack on the **Stacks** page, choose the **Outputs** tab, and open the link for your notebook. See the file **healthcare-and-life-sciences/c-1-protein-folding-quantum-random-walk/protein-folding-qrw.ipynb** for more details.

References:

Wiki: Protein folding

QFold: Quantum Walks and Deep Learning to Solve Protein Folding

Protein Folding – Variational Quantum Eigensolver

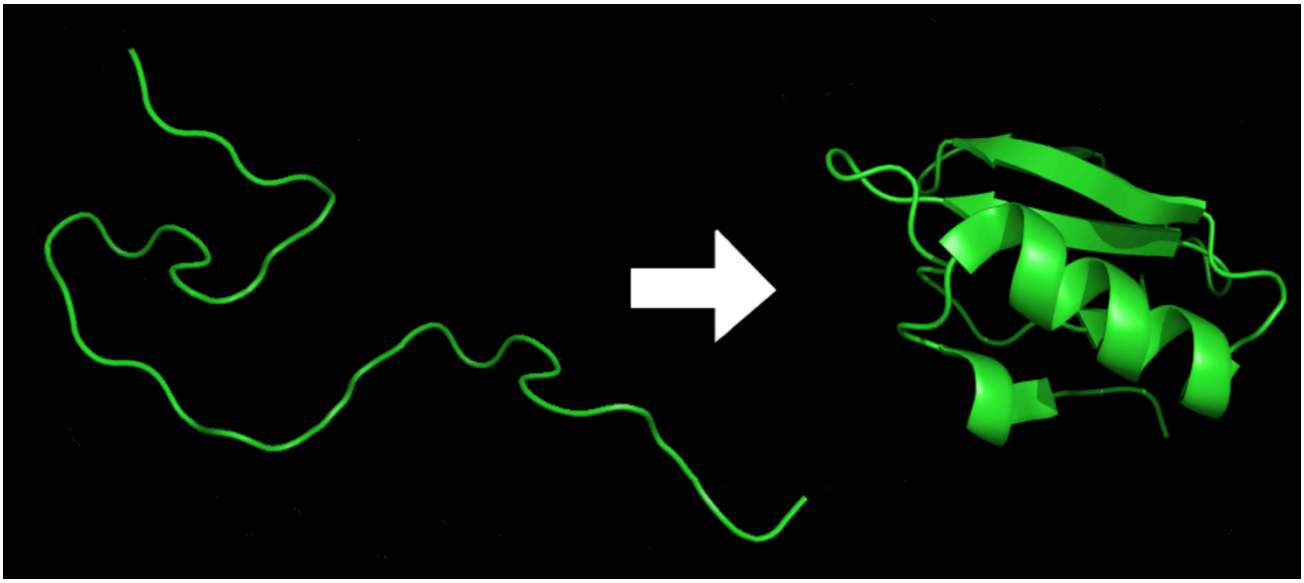
Notebook experimentation

This notebook implements the protein folding using varionational quantum eigensolver (VQE).

Protein folding using VQE

Protein folding is the process by which a protein molecule assumes its three-dimensional shape, which is essential for its proper function. Proteins are made up of a linear chain of amino acids, and their final structure is determined by the sequence of amino acids and the interactions between them.

During protein folding, the linear chain of amino acids folds into a unique three-dimensional structure, which is stabilized by various types of interactions, such as hydrogen bonds, electrostatic forces, and van der Waals forces. The process of protein folding is highly complex and involves multiple stages, including the formation of secondary structures, such as alpha-helices and beta-sheets, and the packing of these structures into a final three-dimensional shape.



Protein before and after folding

VQE is a quantum algorithm that can be used to simulate the behavior of molecules, including proteins. VQE works by approximating the ground state energy of a molecule using a quantum computer, which can be used to predict the properties of the molecule.

To use VQE for protein folding, one would first need to encode the protein structure into a format that can be represented as a quantum state. This can be done using a technique called the qubitization method, which maps the protein structure onto a set of qubits that can be manipulated by the quantum computer.

After deployment, you can go to select the solution's root stack on the **Stacks** page, choose the **Outputs** tab, and open the link for your notebook. See the file **healthcare-and-life-sciences/c-2-protein-folding-variational-quantum-eigensolver/protein-folding-vqe.ipynb** for more details.

References:

Wiki: Protein folding

Protein Folding – Grover's Search

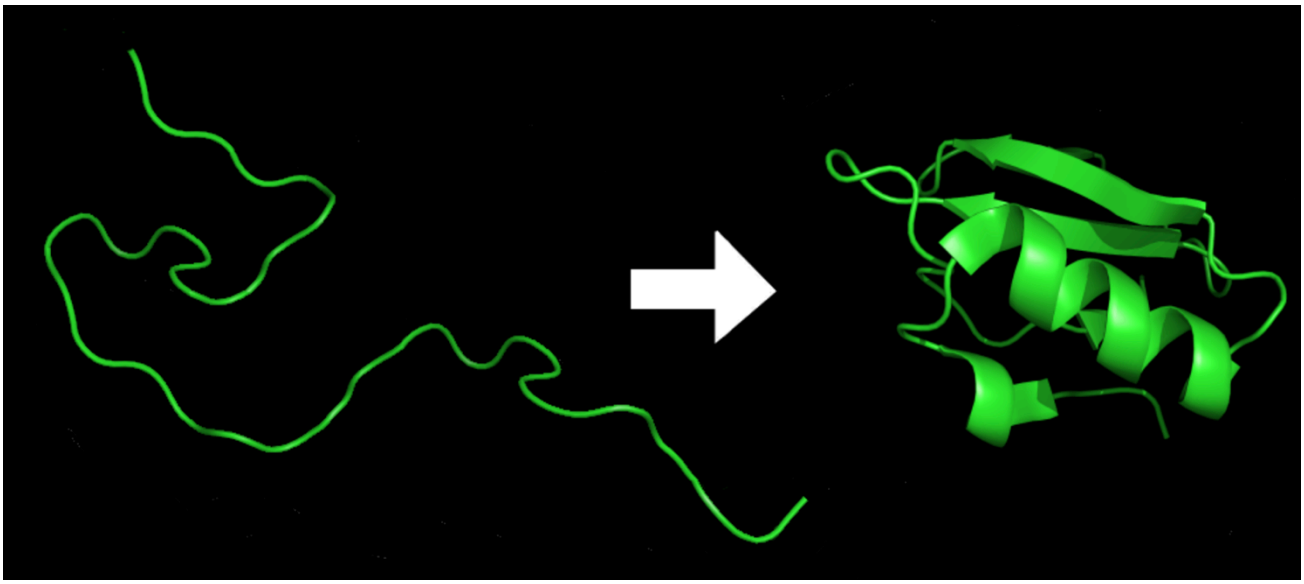
Notebook experimentation

This notebook implements the Quantum Speedup for Protein Structure Prediction in Amazon Braket. This is mainly contributed by Renata Wong and Weng-Long Chang.

Protein folding using Grover's search

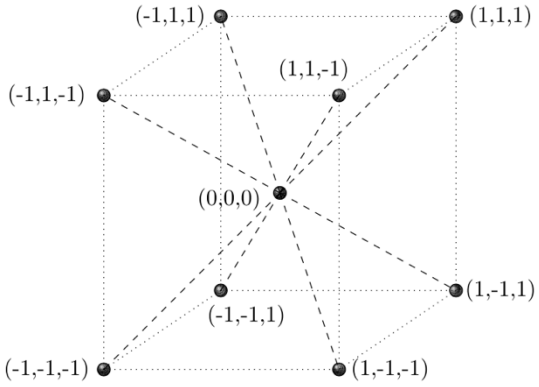
Protein folding is the process by which a protein molecule assumes its three-dimensional shape, which is essential for its proper function. Proteins are made up of a linear chain of amino acids, and their final structure is determined by the sequence of amino acids and the interactions between them.

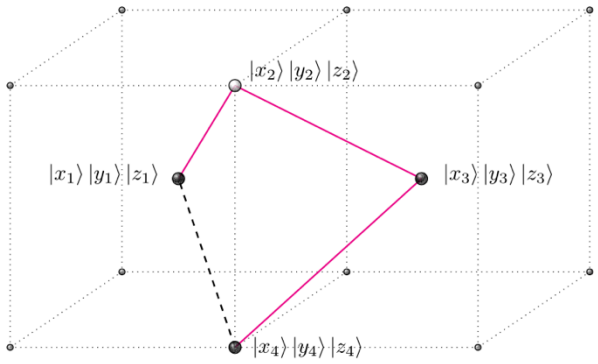
During protein folding, the linear chain of amino acids folds into a unique three-dimensional structure, which is stabilized by various types of interactions, such as hydrogen bonds, electrostatic forces, and van der Waals forces. The process of protein folding is highly complex and involves multiple stages, including the formation of secondary structures, such as alpha-helices and beta-sheets, and the packing of these structures into a final three-dimensional shape.



Protein before and after folding

In this work, a fast quantum algorithm based on Grover's search is proposed. The protein structure prediction problem is studied in three-dimensional hydrophobic-hydrophilic model on body-centered cubic lattice. The results show the quadratic speedup over its classical counterparts.





Body-centered cubic lattice

Grover's algorithm is a quantum algorithm that can be used to search an unsorted database of N items in O(sqrt(N)) time. This is a significant speedup compared to classical algorithms, which require O(N) time to search an unsorted database.

After deployment, you can go to select the solution's root stack on the **Stacks** page, choose the **Outputs** tab, and open the link for your notebook. See the file **healthcare-and-life-sciences/c-3-protein-folding-grover-search/protein-folding-gs.ipynb** for more details.

References:

Wiki: Protein folding

QFold: Quantum Walks and Deep Learning to Solve Protein Folding

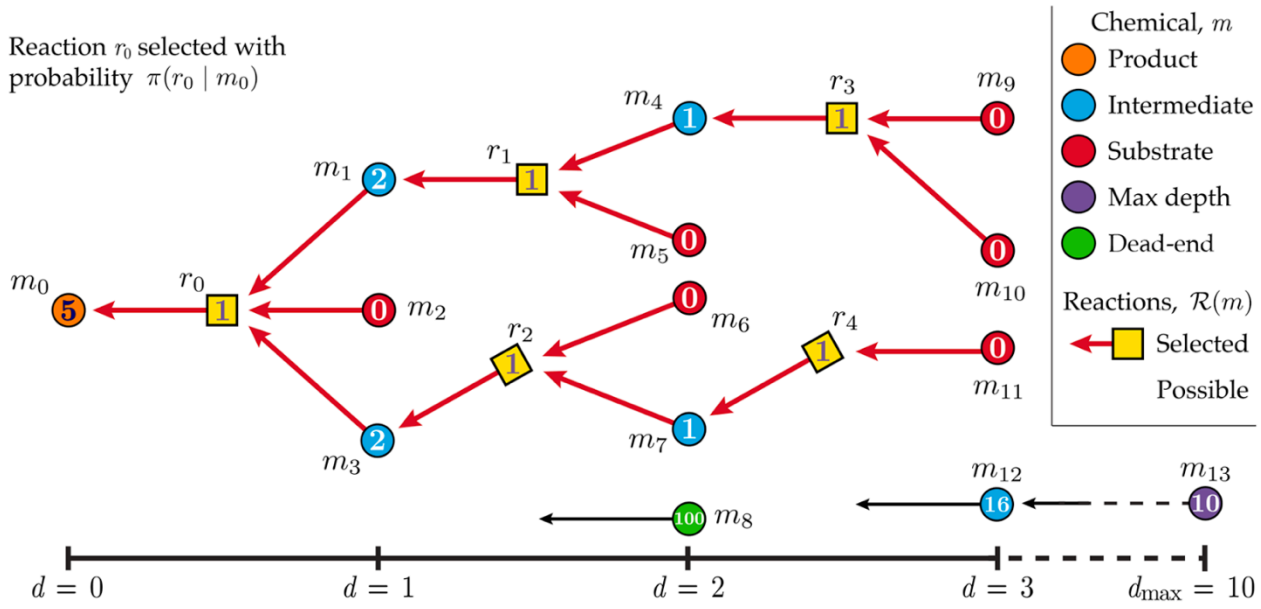
Retrosynthetic Planning – Quantum Reinforcement Learning

Notebook experimentation

This notebook implements the retrosynthetic planning using quantum reinforcement learning. This is mainly contributed by Xiangyu Li inspired by the publication Learning Retrosynthetic Planning through Simulated Experience.

Retrosynthetic planning using quantum reinforcement learning

Retrosynthetic planning is a strategy for designing efficient synthetic routes for the preparation of complex organic molecules. It involves breaking down a target molecule into simpler precursor molecules, and then identifying the synthetic pathways that can be used to synthesize those precursors. It is an essential strategy for designing efficient synthetic routes for complex organic molecules.



A retrosynthetic analysis example

The quantum reinforcement learning for retrosynthetic planning involves encoding the target molecule and the available synthetic transformations as quantum states. The quantum reinforcement learning algorithm can then use these states to learn the optimal sequence of synthetic transformations to reach the target molecule from a set of starting materials. The algorithm can learn from a database of known reactions, as well as from trial and error in a virtual environment, to optimize the synthesis process.

After deployment, you can go to select the solution's root stack on the **Stacks** page, choose the **Outputs** tab, and open the link for your notebook. See the file **healthcare-and-life-science/d-1-retrosynthetic-planning-quantum-reinforcement-learning/retrosynthetic-planning-qrl.ipynb** for more details.

References:

Learning Retrosynthetic Planning through Simulated Experience.

Additional resources

AWS services

- Amazon SageMaker
- Amazon Braket
- Amazon S3
- Amazon EventBridge
- AWS Lambda
- AWS IAM

Uninstall the solution

To uninstall Quantum Computing Exploration for Drug Discovery on AWS, you must delete the AWS CloudFormation stack.

You can use either the AWS Management Console or the AWS Command Line Interface (CLI) to delete the CloudFormation stack.

Using the AWS Management Console

1. Sign in to the AWS CloudFormation console.
2. Select this solution's installation stack.
3. Choose **Delete**.

Using AWS Command Line Interface

Determine whether the AWS Command Line Interface (AWS CLI) is available in your environment. For installation instructions, see What Is the AWS Command Line Interface in the *AWS CLI User Guide*. After confirming that the AWS CLI is available, run the following command.

```
$ aws cloudformation delete-stack --stack-name
      <installation-stack-name>
```

Source code

Visit our GitHub repository to download the source code for this solution. The Quantum Computing Exploration for Drug Discovery on AWS template is generated using the AWS Cloud Development Kit (AWS CDK) (CDK). Refer to the README.md file for additional information.

Contributors

- Aoyu Zhang
- Audrey Wong
- Fan Su
- Hubin Cui
- Jagadeesh Pusapadi
- Mengxin Zhu
- Nihir Chadderwala
- Renata Wong
- Roberto Campos
- Weng-Long Chang
- Xiangyu Li
- Yong Liu

Revisions

Date	Change
May 2022	Initial release
June 2022	Release v1.0.1: bug fixes. For more informati on, refer to the CHANGELOG.md file in the GitHub repository.
June 2023	Release v1.1.0: 1. Restructure the architecture, decouple the user notebook code and the underlying system code 2. Simplify the architecture design and eliminate the batch and QuickSight functions 3. Add the SNS mail subscription function to ensure that you can get the job running results in time

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents AWS current product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. AWS responsibilities and liabilities to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

AWS Glossary

For the latest AWS terminology, see the AWS glossary in the *AWS Glossary Reference*.