# qIDS: Sistema de Detecção de Ataques baseado em Aprendizado de Máquina Quântico Híbrido

**Diego Abreu**[1], **Christian R. Esteve Rothenberg**[2], **Antônio Abelém**[1]

[1]Universidade Federal do Pará (UFPA)

[2]Universidade Estadual de Campinas (UNICAMP)

***Abstract.*** *The rise of quantum utility in the realm of quantum computing presents not just challenges but also significant opportunities for enhancing network security. This paradigm shift in computational capabilities allows for the development of advanced solutions to counteract the rapidly evolving nature of network attacks. Capitalizing on this technological advancement, this work introduces qIDS, an Intrusion Detection System (IDS) that innovatively integrates quantum and classical computing approaches. qIDS leverages Quantum Machine Learning (QML) techniques to effectively learn network behaviors and identify malicious activities. By conducting comprehensive experimental evaluations on public datasets, we demonstrate the proficiency of qIDS in attack detection, excelling in both binary and multiclass classification tasks. Our findings reveal that qIDS competes favorably with classical Machine Learning methods, highlighting the potential of quantum-enhanced cybersecurity solutions in the era of quantum utility.*

***Resumo.*** *A ascensão da utilidade quântica no campo da computação quântica apresenta não apenas desafios, mas também oportunidades para aprimorar a segurança de redes. Esta mudança de paradigma nas capacidades computacionais permite o desenvolvimento de soluções avançadas para contrapor a rápida evolução dos ataques de rede. Aproveitando este avanço tecnológico, este trabalho apresenta o qIDS, um Sistema de Detecção de Intrusão (IDS) que integra de forma inovadora abordagens de computação quântica e clássica. O qIDS utiliza técnicas de Aprendizado de Máquina Quântico (QML) para aprender efetivamente os comportamentos da rede e identificar atividades maliciosas. Ao realizar avaliações experimentais abrangentes em conjuntos de dados públicos, evidenciou-se a competência do qIDS na detecção de ataques, destacando-se, tanto em tarefas de classificação binária quanto multiclasse. Nossos resultados revelam que o qIDS compete favoravelmente com métodos de Aprendizado de Máquina clássicos, destacando o potencial das soluções de cibersegurança aprimoradas por tecnologia quântica na era da utilidade quântica.*

## 1. Introdução

Os recentes avanços na pesquisa e desenvolvimento da computação quântica marcam a entrada na era da utilidade quântica (*quantum utility*) [Kim et al. 2023], um estágio significativo na evolução das tecnologias quânticas. Neste estágio, computadores quânticos já demonstram a capacidade de realizar cálculos confiáveis e eficientes, ultrapassando os limites dos métodos clássicos de força bruta e fornecendo soluções exatas

para problemas complexos. Embora ainda não se tenha atingido a *quantum supremacy* [Boixo et al. 2018], o ponto em que um computador quântico pode resolver problemas inalcançáveis para um computador clássico em um tempo razoável, este progresso atual representa um passo fundamental em direção a aplicações práticas robustas em pesquisa, abrindo caminho para um futuro promissor no uso das tecnologias quânticas.

Nesse contexto, as tecnologias quânticas oferecem a possibilidade de aprimorar a segurança das redes, ao integrar a computação quântica com sistemas de Detecção de Intrusão (*Intrusion Detection System* - IDS) e Prevenção de Intrusão (*Intrusion Prevention System* - IPS). Esse avanço pode ser alcançado por meio do uso de técnicas de Aprendizado de Máquina Quântico (*Quantum Machine Learning* - QML) [Cerezo et al. 2022], permitindo a detecção mais precisa de atividades suspeitas na rede, bem como a análise de grandes volumes de dados, tornando-se fundamental em um ambiente de segurança cibernética em constante evolução.

Um desafio importante está relacionado à capacidade atual dos dispositivos quânticos, conhecidos como dispositivos NISQ (*Noisy Intermediate-Scale Quantum*) [De Luca 2022]. Essas limitações incluem restrições quanto à quantidade de qubits (bits quânticos) disponíveis, à complexidade dos circuitos quânticos que podem ser implementados (profundidade e portas lógicas quânticas disponíveis) e à capacidade de manter a coerência quântica ao longo do tempo (devido à ruídos e a própria natureza dos qubits) [Abelém et al. 2020]. Além disso, a falta de mecanismos robustos de correção de erros em sistemas NISQ também representa um obstáculo significativo [Torlai and Melko 2020]. Portanto, é importante que qualquer proposta de IDS quântico leve em consideração esses fatores para garantir sua aplicabilidade efetiva no cenário atual da computação quântica [De Luca 2022].

Neste trabalho é proposto o qIDS, um Sistema de Detecção de Ataques Baseado em Aprendizado de Máquina Quântico Híbrido. O objetivo principal é criar um sistema adaptável para uso em equipamentos NISQ, superando as limitações inerentes à computação quântica atual. Para alcançar essa meta, nossa abordagem se baseia em técnicas híbridas de Aprendizado de Máquina Quântico, que aproveitam simultaneamente as capacidades da computação clássica e quântica. Para avaliar o desempenho do qIDS, foram conduzidos uma série de experimentos utilizando conjuntos de dados de segurança de rede disponíveis publicamente. Foram comparados três métodos de QML em nosso sistema em termos de detecção de ataques (classificação binária) e identificação de ataques específicos (classificação multiclasse). Além disso, os resultados obtidos com abordagens de QML foram comparados com métodos de Aprendizado de Máquina (AM) clássico. Os resultados experimentais fornecem evidências empíricas da eficácia das técnicas de QML em aprimorar as capacidades de detecção de ataques à rede e apontam para a viabilidade de implementação em sistemas NISQ. As principais contribuições deste trabalho são:

- Desenvolvimento do qIDS, um sistema de detecção de ataques às redes baseado em QML, que utiliza de forma híbrida a computação quântica e computação clássica.
- Apresentação do funcionamento do qIDS mediante a aplicação de três técnicas distintas de QML, seguida de uma avaliação do desempenho de cada abordagem.
- Implementação do qIDS em sistemas NISQ e avaliação de diferentes configurações de circuitos quânticos no desempenho do sistema.

O restante do trabalho está organizado da seguinte maneira: na Seção 2, é abordado o Aprendizado de Máquina Quântico e detalhadas as técnicas de QML utilizadas neste trabalho. A Seção 3 apresenta os trabalhos relacionados, e a Seção 4 apresenta o qIDS, detalhando o seu funcionamento. Na Seção 5 o estudo de caso é descrito, com as configurações e métodos utilizados. Os principais resultados do trabalho e discussões são apresentados na Seção 6. A Seção 7 conclui o trabalho e indica os trabalhos futuros.

## 2. Aprendizado de Máquina Quântico Híbrido

O Aprendizado de Máquina Quântico pode ser entendido como um conjunto de técnicas que combinam princípios da computação quântica (como superposição, interferência e emaranhamento) com técnicas de Aprendizado de Máquina para realizar tarefas como classificação, regressão e agrupamento de dados [Cerezo et al. 2022]. Neste trabalho, nosso foco são as abordagens de QML híbridas, que utilizam ambas computação quântica e computação clássica para a criação dos modelos de aprendizagem.

### 2.1. Classificador Quântico Variacional (VQC)

O Classificador Quântico Variacional (*Variational Quantum Classifier* - VQC) [Havlíček et al. 2019] é um algoritmo de QML que combina circuitos quânticos com técnicas de otimização clássica para realizar tarefas de classificação. A Figura 1 apresenta o funcionamento do VQC. O VQC explora e compara as diferenças entre estados quânticos, que dependem de um conjunto de parâmetros. Esses estados podem ser preparados usando um circuito quântico parametrizado, no qual portas quânticas são definidas com parâmetros ajustáveis. Assim, o VQC emprega um circuito quântico, que pode ser otimizado com base nos dados de treinamento para aprender o limite de decisão ideal entre diferentes classes.

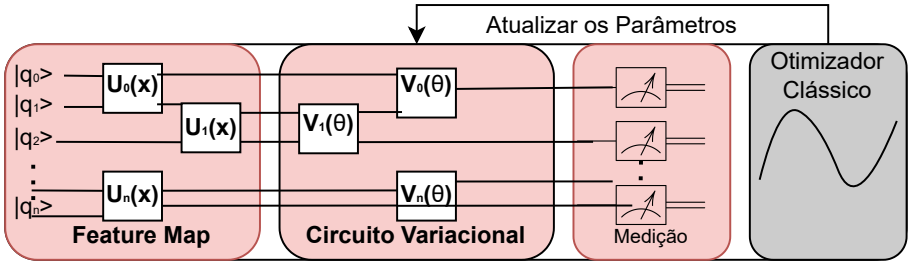



**Figura 1. Funcionamento do Classificador Quântico Variacional.**

No VQC, inicialmente, os dados de entrada (bits clássicos) são codificados em estados quânticos usando um circuito de mapeamento (*feature map)*, como o circuito apresentado na Figura 2. Seja $|\psi_{\text{in}}\rangle$ a representação quântica dos dados de entrada, o *feature map* pode ser expresso como uma operação unitária $U_{\text{FM}}$ atuando sobre $|\psi_{\text{in}}\rangle$, de modo que $|\psi_{\text{out}}\rangle = U_{\text{FM}}|\psi_{\text{in}}\rangle$. Onde $|\psi_{\text{out}}\rangle$ é a representação quântica resultante após a aplicação do *feature map*. A escolha específica do *feature map* influencia a capacidade do VQC em representar as características importantes dos dados de entrada, impactando diretamente o desempenho do modelo gerado.

O segundo passo é a geração de um modelo base através de um circuito quântico variacional conhecido como *ansatz* (circuito teste ou abordagem inicial). A Figura 3 apresenta um exemplo de *ansatz* utilizado para o VQC. Como a Figura 1 destaca, o circuito

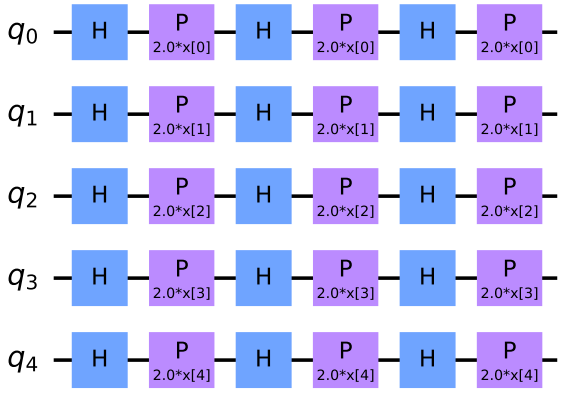



**Figura 2. Exemplo de Circuito *Feature Map*, técnica ZFeatureMap, com 3 camadas de rotação.**

variacional age então sobre os estados codificados, realizando operações e transformações que dependem das portas quânticas de parâmetros ajustáveis, como operações de rotação e emaranhamento, para manipular o estado quântico. Ao realizar a medição do circuito, os qubits colapsam e é obtido valores clássicos de 0 ou 1, os quais são a entrada para parte clássica do QML híbrido.

A terceira etapa consiste na aplicação de um otimizador clássico, o qual ajusta os parâmetros do *ansatz*. O otimizador clássico ajusta iterativamente os parâmetros ajustáveis do circuito variacional, visando minimizar uma função de custo ou perda predefinida. Esse processo de otimização visa encontrar a configuração ideal do circuito variacional que melhor se ajusta ao alvo de classificação desejado.

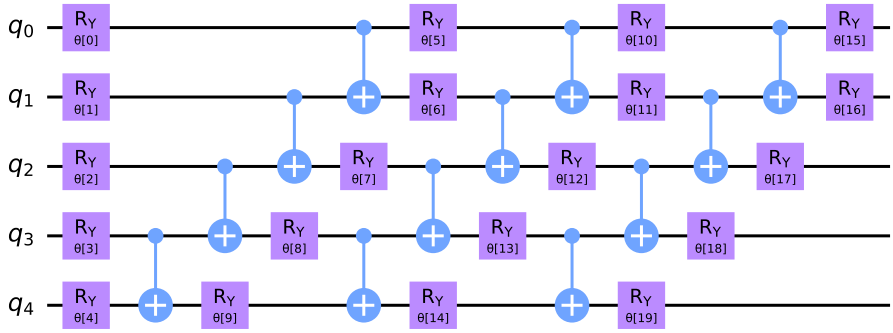



**Figura 3. Exemplo de Circuito Variacional Ansatz.**

## 2.2. Máquina de Vetores de Suporte com Kernel Quântico (QSVM)

A Máquina de Vetores de Suporte com Kernel Quântico (*Quantum Kernel Support Vector Machines* - QSVM) é uma versão quântica do algoritmo clássico de Máquina de Vetores de Suporte (*Support Vector Machines* - SVM) [Mammone et al. 2009]. De forma geral, os métodos de kernel, como o SVM, mapeiam os dados originais para um espaço de *features* onde os dados podem ser melhor entendidos e separados do que originalmente.

A representação matemática para a criação do kernel quântico no QSVM pode ser expressa da seguinte forma: Seja $|\psi_{\text{in},i}\rangle$ a representação quântica do i-ésimo exemplo de treinamento na entrada e $|\psi_{\text{in},j}\rangle$ a representação quântica do j-ésimo exemplo de

treinamento. A função de kernel quântico $K(\mathbf{x}_i, \mathbf{x}_j)$ entre esses dois exemplos é dada por $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \psi_{\text{in},i} | \hat{U}_{\text{kernel}} | \psi_{\text{in},j} \rangle$. Onde $\hat{U}_{\text{kernel}}$ é o operador unitário associado ao kernel quântico. Este operador descreve a transformação quântica realizada no espaço de Hilbert para realizar o mapeamento não linear, permitindo assim que o QSVM lide com dados que não são linearmente separáveis no espaço original [Mammone et al. 2009].

Com o kernel gerado, o QSVM completa o processo de treinamento, ajustando os parâmetros do circuito quântico para otimizar a separação das classes no espaço quântico. Os exemplos de treinamento são mapeados para o espaço quântico através do *feature map* e do operador $\hat{U}_{\text{kernel}}$, resultando em uma matriz de kernel que reflete as relações entre os exemplos. Durante o treinamento, o QSVM identifica um conjunto de vetores de suporte, que são exemplos de treinamento mais relevantes para a definição do hiperplano de separação quântico. Com base nesses vetores e na matriz de kernel, o algoritmo constrói um hiperplano que maximiza a margem entre as classes.

Na fase de teste, quando um novo exemplo não rotulado é apresentado, ele é mapeado para o espaço quântico usando o mesmo *feature map* e $\hat{U}_{\text{kernel}}$. A posição do exemplo em relação ao hiperplano determina a classe à qual será atribuído, seguindo os princípios do SVM. Portanto, o QSVM adapta os conceitos do SVM clássico para o contexto quântico, utilizando transformações quânticas eficientes e um hiperplano quântico para realizar a tarefa de classificação. A escolha do *feature map* e do kernel quântico é crucial para o desempenho do QSVM, permitindo lidar com dados complexos e não linearmente separáveis.

### 2.3. K-Means Quântico - QKM

Similar ao k-means clássico, o k-means quântico (QKM) [Kavitha and Kaulgud 2023] é um algoritmo de aprendizado de máquina não supervisionado cujo objetivo principal é agrupar os dados, identificando e separando grupos diferentes. Na Figura 4, é apresentado o circuito fundamental do QKM, o circuito *swap-test*.

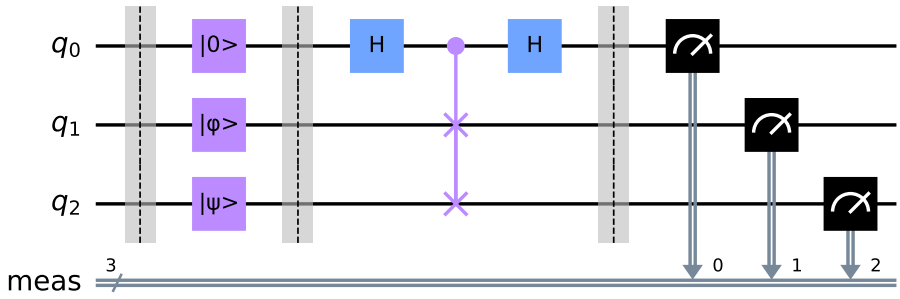



**Figura 4. Circuito *swap-test* usado no quantum k-means.**

No circuito da Figura 4, temos os estados quânticos $|\varphi\rangle$, que representam o centroide do *cluster* (grupo) no conjunto de dados escolhido e $|\Psi\rangle$ como o novo ponto de dados usado para encontrar a qual *cluster* ele pertence. Para calcular a distância entre esses dois vetores, é utilizado um qubit auxiliar no estado $|0\rangle$. Uma porta de Hadamard ($H$) é aplicada ao qubit $|0\rangle$ para colocá-lo em um estado superposição. Usando a porta CX (*control not*) é calculada a distância entre o estado $|\varphi\rangle$ e o estado entrelaçado $|\Psi\rangle$ do qubit auxiliar $|0\rangle$. Após determinar a distância entre os estados $|\varphi\rangle$ e $|\Psi\rangle$ e o centroide,

utiliza-se o circuito do QKM para atribuir cada estado ao seu *cluster* mais próximo. Etapas iterativas são realizadas atualizando os novos centroides de cada *cluster* até que o critério de parada seja satisfeito.

O QKM possui três sub-rotinas: 1) cálculo da distância usando o circuito *swap-test*, 2) atualização do *cluster* e 3) atualização do centroide. O cálculo da distância entre 2 estados é realizado por meio da fórmula Dist $= d^2 r_{\text{max}}^4 (2P(|0\rangle - 1))$. Onde $d$ são pontos de dados não nulos de entrada, $rmax$ é limite máximo de qualquer característica nos dados e $P(|1\rangle) = \frac{1}{2} - \frac{1}{2}|\langle\psi|\phi\rangle|^2$. Assim, a função de sobreposição fornece uma base para realizar tarefas de agrupamento. Para formar o *cluster* final, após cada cálculo de distância, a atualização do *cluster* e do centroide é realizada de forma iterativa. Estas três fases são executadas até que a posição do centroide do *cluster* não mude, indicando convergência.

## 3. Trabalhos Relacionados

Diversas pesquisas têm abordado o uso do Aprendizado de Máquina Quântico [Cerezo et al. 2022]. Nessa seção destacamos os trabalhos que buscam utilizar o QML no contexto de segurança de redes. A Tabela 1 apresenta uma comparação da nossa pesquisa com os trabalhos relacionados.

**Tabela 1. Comparação entre Nosso Trabalho e Trabalhos Relacionados.**

| | **Técnicas** | **NISQ** | **Ataques** | **Alvo** | **Base de Dados** |
|---|---|---|---|---|---|
| qIDS | VQC, QSVM QKM | ✓ | Vários (21) | Binário Multiclasse | UNSW-NB15, CIC-17, TON-IOT |
| [Said 2023] | QSVM | X | DDoS | Binário | CIC-DDoS2019 |
| [Gong et al. 2022] | VQC + NN | X | Vários (4) | Binário | KDD CUP99 |
| [Kalinin 2023] | QSVM + NN | X | DoS | Binário | Base própria |

Said et al (2023). [Said 2023] explora a aplicação de um modelo de QSVM para detectar Ataques de Negação de Serviço Distribuído (*Distributed Denail of Service* - DDoS) em *smart microgrids*. O estudo avalia o modelo de QML usando uma versão reduzida (por amostragem) do conjunto de dados CIC-DDoS2019 [Sharafaldin et al. 2019], que inclui dados tanto de ataques DDoS, quanto do comportamento normal da rede. Os resultados demonstram a superioridade do modelo de QML em comparação com a abordagem clássica, SVM.

No artigo de Gong et al. [Gong et al. 2022], é proposta a aplicação de uma Rede Neural (RN) em conjunto com o VQC para a detecção de ataques de rede. Os autores aplicam o modelo criado a um subconjunto de recursos reduzidos do conjunto de dados KDD CUP99 [Elkan 2000], incorporando uma abordagem de equilíbrio de classes para aumentar a precisão da classificação. No entanto, embora este estudo sirva como uma ilustração do uso do VQC com RN, o conjunto de dados KDD CUP99 está desatualizado e pode não refletir com precisão os comportamentos de rede atuais. De outro modo, Kalinin e Krundyshev [Kalinin and Krundyshev 2023] propõem um modelo de QSVM com RN para detectar ataques de rede. O modelo gerado é aplicado em uma base de dados criada pelos autores, e comparado com outras técnicas como o SVM e RN convolucionais.

Como a Tabela 1 destaca, nossa pesquisa se difere dos trabalhos relacionados por apresentar uma análise abrangente da aplicação de várias técnicas de QML (VQC, QSVM e QKM), tanto para a detecção (cenário binário) quanto para a identificação de ataques (cenário multiclasse). Além disso, nosso trabalho realiza testes em três conjuntos de dados de referência em segurança de rede, contendo uma grande diversidade de ataques à rede. Outro ponto importante é que os trabalhos relacionados implementaram as suas soluções utilizando simuladores de computação quântica sem ruído, os quais não conseguem representar corretamente os sistemas NISQ atuais. Em nosso trabalho, nossa proposta é implementada também utilizando *backends* com ruído, os quais representam de forma mais próxima os equipamentos NISQ atuais.

## 4. qIDS: Sistema de Detecção de Ataques baseado em Aprendizado de Máquina Quântico Híbrido

Neste trabalho é proposto um sistema de detecção de ataques à rede baseado em Aprendizado de Máquina Quântico Híbrido (qIDS). A proposta é aplicar a técnicas QML Híbridas, que utilizam computação quântica e computação clássica, para realizar a detecção de ataques à rede. A proposta tem o funcionamento apresentado no fluxograma da Figura 5.

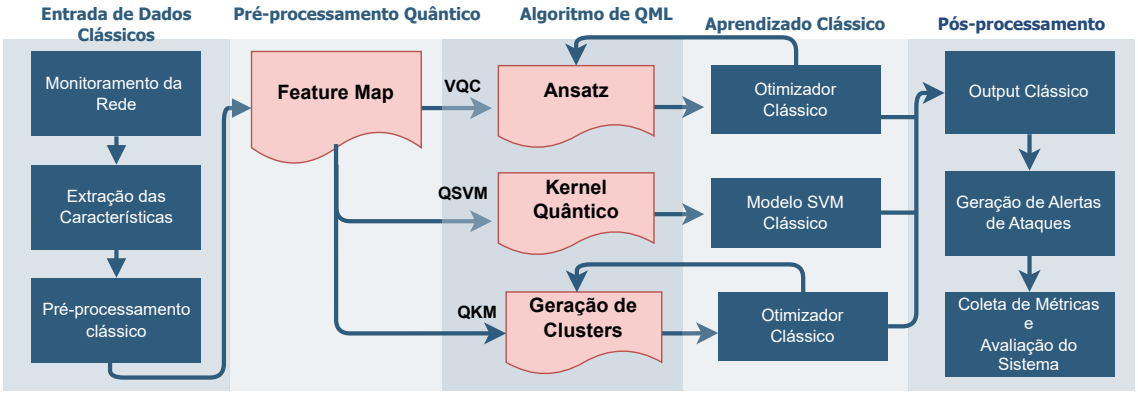



**Figura 5. Fluxograma de funcionamento do qIDS.**

Como a Figura 5 ilustra, o qIDS inicia com o monitoramento da rede e coleta dos dados, a partir dos quais são extraídas características (*features*). Após isso, é realizado o pré-processamento nos dados para normalização, tratamento de valores ausentes e outras técnicas que visam adequar os dados para o sistema. Em seguida, ocorre o mapeamento dos dados clássicos para os estados quânticos, através do *feature map*. O processo do QML Híbrido é então iniciado, com a parte quântica e clássica de acordo com a técnica de QML. No caso do VQC, é gerado o circuito quântico variacional antaz o qual tem seus parâmetros ajustados por um otimizador clássico. No QSVM, é gerado um kernel quântico o qual é usado para treinar o modelo de predição clássico. Já no QKM, o processo quântico usa o circuito *swap-test* para calcular e gerar os *clusters*, e um otimizador clássico ajusta os parâmetros do KQM.

O resultado final do processo é uma saída clássica, uma interpretação dos resultados quânticos que é utilizada para a geração de alertas de ataques. Com base nas análises realizadas, o qIDS gera alertas de ataques sempre que padrões suspeitos são identificados, contribuindo para a segurança da rede. É importante ressaltar que nessa abordagem outros QML híbridos também podem ser utilizados, respeitando as particularidades de cada

método. O qIDS pode ser implementado então parte em um sistema clássico e parte em um sistema quântico. Devido ausência de computadores quânticos comerciais, o processo quântico pode ser realizado por meio de *backends* remotos, como simuladores quânticos disponíveis em supercomputadores[1] ou computadores quânticos privados, como os disponíveis pela IBM[2].

## 5. Estudo de Caso

Para avaliar o sistema proposto, foi realizado um estudo de caso usando três bases de dados de referência em segurança de rede: UNSW-NB15 [Moustafa and Slay 2015], CICIDS17 [Sharafaldin et al. 2018], TON-IOT [Booij et al. 2021]. Essas bases de dados contêm dados de tráfego de rede, tanto de comportamento normal quanto de diferentes tipos de ataques, e estão entre as mais utilizadas no contexto de detecção e classificação de ataques usando técnicas de Aprendizado de Máquina.

Neste estudo de caso, foi realizado uma comparação do desempenho entre as técnicas de QML e as de AM clássico. Para tal, foi selecionado técnicas amplamente utilizadas no contexto de detecção de ataques, como *Random Forest* (RF), *k-Nearest Neighbors* (kNN) e SVM. Essas técnicas são conhecidas por sua simplicidade de treinamento e aplicação, servindo como um comparativo eficaz entre o QML e métodos tradicionais de AM.

### 5.1. Configuração dos Experimentos

Nesta subseção, é descrita a configuração experimental utilizada para avaliar o desempenho do sistema qIDS. Três modelos de Aprendizado de Máquina Quântico foram empregados: VQC, QSVM e QKM, os quais foram implementados utilizando o framework qiskit[3]. Para estes modelos, os hiperparâmetros específicos utilizados são apresentados na Tabela 2. Foram selecionados 4 tipos de *Feature Maps*, de circuito *ansatz* e de otimizadores clássicos, dentre os disponíveis no qiskit e também, frequentemente, utilizados em QML [Buonaiuto et al. 2023].

Neste estudo de caso, além dos modelos quânticos, foi realizado uma comparação com modelos clássicos de Aprendizado de Máquina, implementados usando o framework scikit-learn[4]. Para os modelos clássicos, incluindo kNN, SVM e Random Forest, os hiperparâmetros foram otimizados utilizando *grid search*. No modelo kNN, os parâmetros incluem o número de vizinhos (variando de 3 a 15), métricas (euclidiana ou manhattan), pesos (uniforme ou distância), algoritmo (auto, ball_tree, kd_tree, brute) e tamanho da folha (entre 10 e 100). Para o SVM, os parâmetros consistiam em C (0.1, 1, 10), gamma (escala ou auto) e kernel (linear, poli, rbf, sigmóide). Já o Random Forest foi ajustado com profundidade máxima (de 3 a 20), número de estimadores (de 10 a 1000) e critério (gini, entropia ou log_loss). A seleção destes parâmetros foi baseada em práticas comuns de busca de hiperparâmetros [Ali et al. 2023], buscando uma avaliação abrangente e comparativa do sistema qIDS, evidenciando as vantagens e limitações de cada método de aprendizado de máquina no contexto da detecção de ataques de rede.

[1]https://atos.net/en/lp/myqlm
[2]https://www.ibm.com/quantum
[3]https://www.ibm.com/quantum/qiskit
[4]https://scikit-learn.org/stable/

**Tabela 2. Hiperparâmetros dos Modelos Quânticos.**

| Feature Maps | Ansatz | Otimizador |
|---|---|---|
| PauliFeatureMap | EfficientSU2 | COBYLA |
| RawFeatureVector | ExcitationPreserving | ADAM |
| ZFeatureMap | RealAmplitudes | SPSA |
| ZZFeatureMap | TwoLocal | GradientDescent |

Nos experimentos com computação quântica, um desafio significativo é a transpilação (*transpile*) de circuitos quânticos. Embora os simuladores, como o QASM, suportem uma grande variedade de portas lógicas quânticas, os computadores quânticos reais são limitados a um conjunto restrito de portas quânticas fisicamente implementáveis. Essa limitação exige que os circuitos quânticos, inicialmente projetados para um ambiente de simulação, sejam adaptados ou transpilados para serem compatíveis com as capacidades do hardware quântico real. A transpilação envolve a reestruturação do circuito quântico original, convertendo-o para um formato que se alinhe com as portas e os processos disponíveis no equipamento quântico específico. Esse processo é crucial para garantir que os experimentos e as operações quânticas possam ser realizados efetivamente no hardware disponível, respeitando suas restrições físicas e operacionais. A Figura 6 ilustra um exemplo de um circuito de Feature Map (Figura 2) após o processo de transpilação, demonstrando as adaptações necessárias para sua implementação em um ambiente quântico real, como a substituição da porta de Hadamard (H) por portas Rz e $\sqrt{X}$.

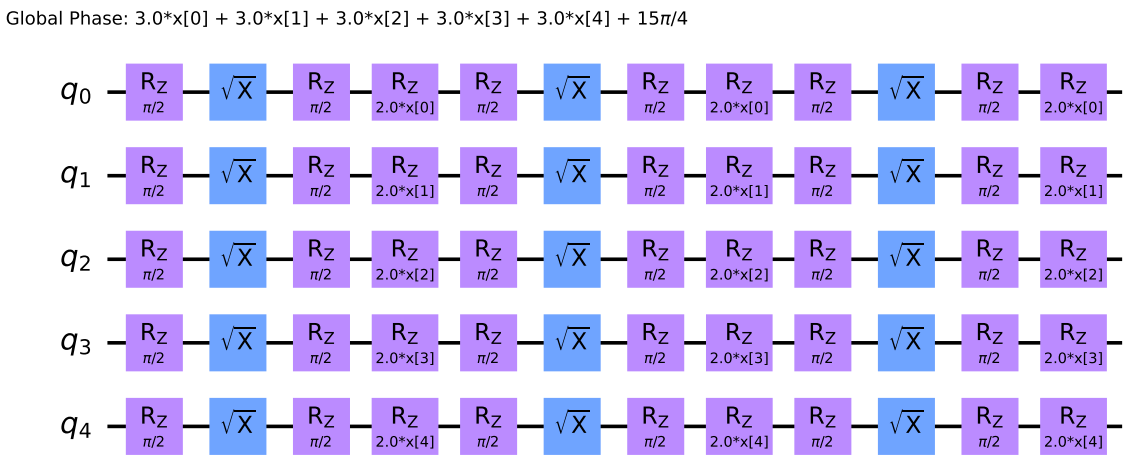


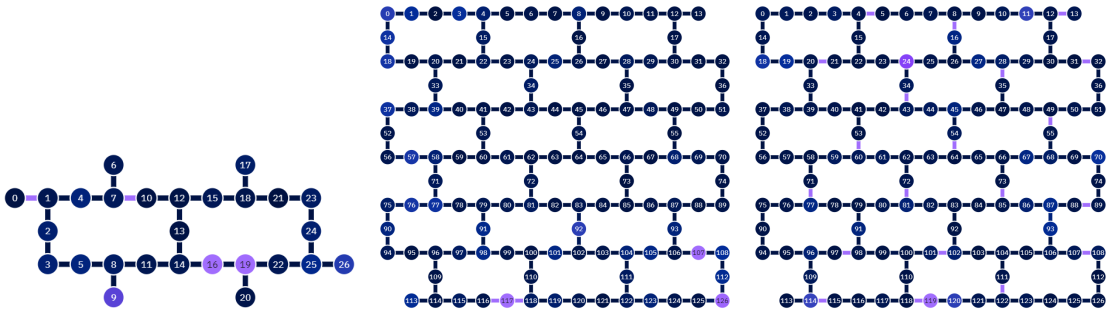

**Figura 6. Circuito de Feature Map após a transpilação.**

## 6. Resultados

Nessa seção são apresentados os resultados do estudo de caso. Primeiramente são apresentados os resultados do qIDS em em diferentes configurações de *Backends* NISQ. Após isso, são apresentados os resultados tanto para a detecção de ataques (classificação binária) quanto para a identificação dos ataques (classificação multiclasse).

### 6.1. Resultados do qIDS em diferentes configurações de *Backends* NISQ

Para o estudo de caso, foram testados 4 diferentes *backends*. O primeiro *backend* é o *QASM*, um simulador de computação quântica sem ruído disponível no qiskit[5]. Os outros 3 são *backends* que utilizam as configurações (modelo de ruído, portas lógicas quânticas e topologia) de computadores quânticos reais. Nesse caso, foram utilizados os dados do *IBM_CAIRO*, *IBM_KYOTO* e *IBM_BRISBANE*. A Figura 7 apresenta a topologia (posição física dos qubits) e o ruído por qubit e conexão (quanto mais escuro a cor menor o ruído) de cada *Backend* usado. Pode-se observar na Figura 7 que o IBM_Cairo possui 27 qubits e uma topologia diferente do IBM_Kyoto e IBM_Brisbane, os quais são computadores quânticos de 127 qubits. Para ilustrar os resultados obtidos, a Tabela 3 apresenta



(a) IBM Cairo (b) IBM Kyoto (c) IBM Brisbane

**Figura 7. Topologia dos *Backends* de Computadores Quânticos da IBM.**

o desempenho, medido pela métrica F1 Score, dos QML em cada *backend* para o caso de detecção binária de ataques. Esta métrica foi escolhida por ser um indicador equilibrado entre precisão e recall, essencial em contextos de segurança onde tanto a detecção de ataques quanto a minimização de falsos positivos são importantes.

**Tabela 3. F1 Score dos QML em cada *Backend* nas três base de dados (para o caso binário)**

| UNSW-NB15 | QASM | IBM_CAIRO | IBM_KYOTO | IBM_BRISTANE |
|---|---|---|---|---|
| VQC | 88.56% | 87.21% | **88.99%** | 85.73% |
| QSVM | 89.34% | **87.76%** | 85.12% | 85.88% |
| KQM | 87.45% | 85.33% | **86.78%** | 86.22% |
| CIC-IDS-17 | QASM | IBM_CAIRO | IBM_KYOTO | IBM_BRISTANE |
| VQC | 95.12% | 93.45% | **95.78%** | 92.01% |
| QSVM | 94.67% | 92.89% | **94.23%** | 92.55% |
| KQM | 95.88% | 92.34% | **93.67%** | 91.45% |
| TON-IOT | QASM | IBM_CAIRO | IBM_KYOTO | IBM_BRISTANE |
| VQC | 98.23% | 95.56% | **97.78%** | 94.34% |
| QSVM | 98.67% | **96.12%** | 95.45% | 94.89% |
| KQM | 98.45% | 95.78% | **96.11%** | 94.56% |

[5] $https://qiskit.org/ecosystem/aer/stubs/qiskit\_aer.QasmSimulator.html$

A Tabela 3 oferece uma visão detalhada e comparativa dos resultados obtidos pelos QML em diferentes *backends*, ilustrando como as condições de hardware afetam a eficácia na detecção de ataques em cenários binários. O *backend* QASM, que simula um ambiente de computação quântica sem ruído, apresenta consistentemente os maiores valores de F1 Score em todas as três bases de dados (UNSW-NB15, CIC-IDS-17 e TON-IOT), indicando um desempenho superior. No entanto, os resultados do QASM não foram considerados no contexto mais amplo da pesquisa, uma vez que o objetivo é avaliar o desempenho em sistemas quânticos reais, particularmente em sistemas NISQ, onde o ruído e outras limitações físicas são fatores inerentes e desafiadores. Por isso, a ênfase foi colocada nos resultados dos *backends IBM_CAIRO*, *IBM_KYOTO* e *IBM_BRISBANE*, que representam condições mais realistas e práticas de computação quântica.

Ao observar os resultados nesses *backends*, percebe-se uma variação significativa nos F1 Scores entre eles, refletindo a influência do modelo de ruído, das portas lógicas quânticas e da topologia específica de cada computador quântico real. Esta variação também indica que diferentes QMLs (VQC, QSVM, KQM) reagem de maneira distinta às condições de cada *backends*, evidenciando a importância de selecionar e ajustar os modelos de acordo com as características específicas do hardware quântico utilizado. Os melhores resultados obtidos em cada combinação de modelo e *backends* serão discutidos em detalhes nas subseções seguintes.

### 6.2. Resultados do qIDS: Detecção de Ataques

A Tabela 4 apresenta uma comparação entre os QMLs e métodos tradicionais de AM, SVM, RF, e KNN, para detecção de ataques (classificação binária) em três diferentes conjuntos de dados: UNSW-NB15, CIC-IDS-17, e TON-IOT.

**Tabela 4. Comparação dos QML e AM - métrica F1 Score - classificação binária.**

| | UNSW-NB15 | CIC-IDS-17 | TON-IOT |
|---|---|---|---|
| VQC | **88.99** | **95.78** | **97.78** |
| QSVM | 87.76 | 94.23 | 96.12 |
| KQM | 86.78 | 93.67 | 96.11 |
| SVM | 82.34 | 93.78 | 96.45 |
| DT | 82.67 | 92.45 | 96.23 |
| Knn | **84.89** | 92.01 | 96.67 |

Observando os resultados, é notável que os QMLs, em geral, apresentam desempenhos comparáveis ou superiores aos métodos de AM. Em particular, o VQC destaca-se com os maiores F1 Score nos três conjuntos de dados, indicando uma eficácia significativa na detecção de ataques. Por exemplo, no dataset UNSW-NB15, o VQC atinge 88.99%, enquanto o melhor método de AM, o KNN, alcança 84.89%. Esta tendência é consistente em CIC-IDS-17 e TON-IOT, onde os QMLs superam os métodos de AM, embora por margens variáveis. Uma observação importante é que, embora os QMLs mostrem vantagens em alguns casos, a diferença de desempenho entre eles e os métodos de AM tradicionais não é significativamente grande. Isso sugere que, enquanto os QMLs oferecem benefícios potenciais, especialmente em cenários específicos ou quando ajustados adequadamente ao contexto e às características dos dados, eles ainda não são substancialmente superiores em todos os aspectos.

### 6.3. Resultados do qIDS: Identificação dos Ataques

**Tabela 5. Comparação dos resultados de F1 Score dos QML e métodos de AM para diferentes tipos de ataques da base UNSW-NB15.**

| Ataque | VQC | QSVM | KQM | SVM | RF | Knn |
|---|---|---|---|---|---|---|
| Analysis | 95.34 | 87.89 | 94.12 | 96.45 | 90.23 | 56.78 |
| Backdoor | 75.56 | 77.01 | 88.45 | 90.78 | 92.11 | 49.34 |
| DoS | 76.89 | 78.23 | 79.67 | 92.45 | 82.56 | 76.89 |
| Exploits | 94.23 | 92.78 | 90.45 | 79.01 | 85.89 | 89.45 |
| Fuzzers | 74.12 | 96.34 | 84.67 | 77.56 | 79.78 | 76.23 |
| Generic | 96.01 | 93.45 | 94.89 | 99.34 | 99.01 | 98.45 |
| Reconn | 97.56 | 99.78 | 89.34 | 86.12 | 82.89 | 78.56 |
| Shellcode | 66.45 | 71.67 | 72.34 | 76.23 | 80.12 | 48.67 |
| Worms | 22.78 | 44.89 | 69.45 | 20.56 | 35.78 | 42.89 |

**Tabela 6. Comparação dos resultados de F1 Score dos QML e métodos de AM para diferentes tipos de ataques da base CIC-IDS-17.**

| Ataque | VQC | QSVM | KQM | SVM | RF | Knn |
|---|---|---|---|---|---|---|
| BoT | 81.23 | 96.11 | 98.45 | 94.67 | 92.12 | 98.89 |
| BruteForce | 92.87 | 100.00 | 98.76 | 93.45 | 96.24 | 95.18 |
| DDoS | 86.54 | 100.00 | 98.11 | 94.32 | 99.34 | 100.00 |
| DoS | 89.32 | 88.79 | 99.23 | 99.67 | 98.02 | 88.45 |
| Infiltration | 83.99 | 99.21 | 92.54 | 99.87 | 97.53 | 97.11 |
| PortScan | 90.45 | 94.32 | 94.76 | 97.89 | 97.45 | 99.76 |
| WebAttack | 89.78 | 92.67 | 97.33 | 98.21 | 96.89 | 89.45 |

**Tabela 7. Comparação dos resultados de F1 Score dos QML e métodos de AM para diferentes tipos de ataques da base TON-IOT.**

| Ataque | VQC | QSVM | KQM | SVM | RF | Knn |
|---|---|---|---|---|---|---|
| Backdoor | 95.67 | 91.12 | 76.45 | 79.78 | 89.01 | 78.34 |
| DoS | 98.45 | 92.23 | 88.78 | 85.56 | 85.34 | 79.12 |
| DDoS | 98.12 | 90.45 | 92.01 | 76.89 | 88.45 | 88.23 |
| Injection | 98.89 | 93.67 | 90.56 | 91.34 | 90.23 | 65.78 |
| MITM | 97.56 | 99.01 | 92.45 | 88.78 | 86.34 | 78.67 |
| Password | 97.01 | 90.89 | 90.78 | 86.45 | 87.56 | 87.45 |
| Ransomware | 75.78 | 78.34 | 89.12 | 73.56 | 85.23 | 66.56 |
| Scanning | 82.67 | 88.45 | 88.89 | 94.01 | 82.45 | 90.12 |
| XSS | 84.23 | 84.56 | 91.23 | 84.78 | 80.45 | 86.01 |

As Tabelas 5, 6 e 7 apresentam os resultados do F1 Score para a identificação de ataques (classificação multiclasse) nas bases de dados UNSW-NB15, CIC-IDS-17 e TON-IOT, respectivamente. Na **Tabela 5**, observa-se que os QMLs, como VQC, QSVM e KQM, mostram desempenhos variados em relação aos diferentes tipos de ataques da base UNSW-NB15. Por exemplo, eles se destacam na detecção de ataques como *Analysis* e *Reconnaissance* (Reconn), mas apresentam resultados inferiores em categorias como

*Worms* e *Shellcode* em comparação com métodos de AM como SVM e RF. Isso indica que os QMLs podem ser mais eficazes para certos tipos de ataques, enquanto em outros, os métodos tradicionais mantêm sua superioridade. Na **Tabela 6**, que se concentra na base de dados CIC-IDS-17, nota-se uma tendência semelhante. Os QMLs, especialmente o QSVM e o KQM, atingem pontuações de F1 significativas em ataques como *BruteForce* e *DDoS*, indicando uma alta precisão na identificação desses ataques. No entanto, em ataques como *DoS* e *WebAttack*, os métodos tradicionais de AM, como SVM e RF, mostram desempenhos comparáveis ou até superiores. Por fim, a **Tabela 7** apresenta os resultados para a base TON-IoT. Aqui, os QMLs novamente demonstram forte desempenho em ataques como *DoS*, mas em tipos de ataques como *Ransomware* e *Scanning*, os métodos tradicionais de AM se sobressaem. Isso sugere que, enquanto os QMLs oferecem vantagens significativas em determinados cenários, eles ainda não são universalmente superiores em todas as formas de detecção de ataques.

## 7. Conclusão e Trabalhos Futuros

Este trabalho apresentou o qIDS, um sistema de detecção de ataques baseado em Aprendizado de Máquina Quântico Híbrido, projetado para enfrentar desafios emergentes em cenários de cibersegurança. Por meio de avaliações experimentais conduzidas em diferentes conjuntos de dados públicos, o qIDS demonstrou ser eficaz na detecção de ataques, tanto em classificações binárias quanto multiclasse, exibindo resultados competitivos em comparação com os métodos clássicos de Aprendizado de Máquina.

No atual cenário NISQ e de utilidade quântica, a aplicação do qIDS já demonstra resultados competitivos em comparação com outras técnicas de AM na detecção e identificação de ataques. Entretanto, ainda existem desafios e limitações a serem superados para implementação prática da proposta em ambientes reais de rede, o que continua sendo uma área de pesquisa ativa e abre espaço para trabalhos futuros. Entre os pontos a serem explorados, está a integração do qIDS com IDS clássicos e sistemas IDS que fazem uso de AM clássico, buscando complementar esses sistemas. Além disso, a limitação de hardware quântico disponível, bem como questões de privacidade no processamento dos dados na plataforma da IBM podem ser abordados, avançando o estado da arte.

## Agradecimentos

Este trabalho foi parcialmente financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e pela Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), projeto 2020/04031-1, projeto 2021/00199-8 (CPE SMARTNESS), projeto 2023/00673-7 e projeto 2023/00811-0.

## Referências

Abelém, A., Vardoyan, G., and Towsley, D. (2020). Quantum internet: The future of internetworking. In *Minicursos do XXXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 48–90. SBC.

Ali, T. E., Chong, Y.-W., and Manickam, S. (2023). Machine learning techniques to detect a ddos attack in sdn: A systematic review. *Applied Sciences*, 13(5):3183.

Boixo, S., Isakov, S. V., Smelyanskiy, V. N., Babbush, R., Ding, N., Jiang, Z., Bremner, M. J., Martinis, J. M., and Neven, H. (2018). Characterizing quantum supremacy in near-term devices. *Nature Physics*, 14(6):595–600.

Booij, T. M., Chiscop, I., Meeuwissen, E., Moustafa, N., and Den Hartog, F. T. (2021). Ton_iot: The role of heterogeneity and the need for standardization of features and attack types in iot network intrusion data sets. *IEEE Internet of Things Journal*.

Buonaiuto, G., Gargiulo, F., De Pietro, G., Esposito, M., and Pota, M. (2023). Best practices for portfolio optimization by quantum computing, experimented on real quantum devices. *Nature Scientifc Reports*, 13:19434.

Cerezo, M., Verdon, G., Huang, H.-Y., Cincio, L., and Coles, P. J. (2022). Challenges and opportunities in quantum machine learning. *Nature Computational Science*.

De Luca, G. (2022). A survey of nisq era hybrid quantum-classical machine learning research. *Journal of Artificial Intelligence and Technology*, 2(1):9–15.

Elkan, C. (2000). Results of the kdd'99 classifier learning. *Acm Sigkdd Explorations Newsletter*, 1(2):63–64.

Gong, C., Guan, W., Gani, A., and Qi, H. (2022). Network attack detection scheme based on variational quantum neural network. *The Journal of Supercomputing*, 78(15):16876–16897.

Havlíček, V., Córcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M., and Gambetta, J. M. (2019). Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212.

Kalinin, M. and Krundyshev, V. (2023). Security intrusion detection using quantum machine learning techniques. *Journal of Computer Virology and Hacking Techniques*, 19(1):125–136.

Kavitha, S. and Kaulgud, N. (2023). Quantum k-means clustering method for detecting heart disease using quantum circuit approach. *Soft Computing*, 27(18):13255–13268.

Kim, Y., Eddins, A., Anand, S., Wei, K. X., Van Den Berg, E., Rosenblatt, S., Nayfeh, H., Wu, Y., Zaletel, M., Temme, K., et al. (2023). Evidence for the utility of quantum computing before fault tolerance. *Nature*, 618(7965):500–505.

Mammone, A., Turchi, M., and Cristianini, N. (2009). Support vector machines. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(3):283–289.

Moustafa, N. and Slay, J. (2015). Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). In *2015 military communications and information systems conference (MilCIS)*, pages 1–6. IEEE.

Said, D. (2023). Quantum computing and machine learning for cybersecurity: Distributed denial of service (ddos) attack detection on smart micro-grid. *Energies*, 16(8):3572.

Sharafaldin, I., Lashkari, A. H., and Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, 1:108–116.

Sharafaldin, I., Lashkari, A. H., Hakak, S., and Ghorbani, A. A. (2019). Developing realistic distributed denial of service (ddos) attack dataset and taxonomy. In *2019 International Carnahan Conference on Security Technology (ICCST)*, pages 1–8. IEEE.

Torlai, G. and Melko, R. G. (2020). Machine-learning quantum states in the nisq era. *Annual Review of Condensed Matter Physics*, 11:325–344.