Chair for Design Automation
TUM School of Computation, Information and Technology
Technical University of Munich

The Munich Quantum Toolkit (MQT)

Design Automation Tools and Software for Quantum Computing

Robert Wille and Team

Abstract

Quantum computers are becoming a reality. But designing applications for these devices requires automated, efficient, and user-friendly software tools that cater to the needs of end-users, engineers, and physicists at every level of the design flow. The Munich Quantum Toolkit (MQT) is an open-source collection of design automation tools and software for quantum computing. This flyer provides an overview of the provided solutions. For each step in the design flow, numbered nodes indicate the respectively available software repositories and tools, which are summarized on the back of this flyer. All code is hosted publicly on GitHub under a permissive license. Packages are available from PyPI.

Data Structures / Core Methods 14 15 16

To tackle the complexity of important design tasks, the MQT utilizes efficient data structures (e.g., for the representation and manipulation of quantum states and operations) as well as dedicated core methods (e.g., allowing to realize optimal methods) including:

- Decision Diagrams
- Tensor Networks
- ZX-Calculus
- SAT/SMT Solvers
- Machine Learning
- Heuristics

MUNICH QUANTUM TOOLKIT

For performance reasons, most tools are implemented in C++ with convenient Python bindings and compatibility to tools such as Qiskit.

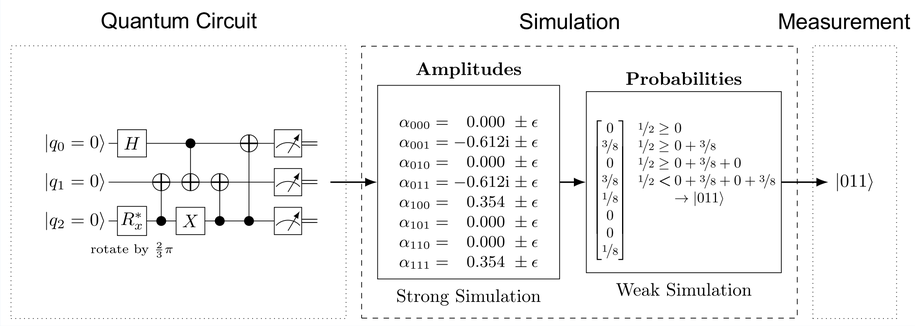
Application 1 2

- Quantum program benchmark suite
- Resource estimation
- Workflows for deriving quantum solutions to classical problems

Application

Simulation 3 4

- Efficient classical circuit simulation based on decision diagrams, supporting unitary, hybrid, and sampling approaches
- Simulation of open quantum systems and noisy circuits using tensor network methods



Simulation

Compilation 5 6 7 8 9 10

- Compiler infrastructure (MLIR / QIR)
- Technology-specific compiler passes for
  - Superconducting platforms
  - Neutral Atom platforms
  - Trapped Ion platforms
- Compiler and scheduling optimization
- Multi-level (qudit) compilation

Compilation

Verification 11 12

- Equivalence checking of quantum circuits
- Verification of compilation results
- Debugging of quantum programs

Verification

Error Correction 13

- Decoding algorithms for color codes and QLDPC codes
- Fault-tolerant gadget optimization
- Lattice surgery compilation

Error Correction

Hardware

Use it

Read the Docs
mqt.readthedocs.io

Open-source Implementations
github.com/munich-quantum-toolkit

```
Shell
$ (uv) pip install mqt.<tool>
```

Contact: robert.wille@tum.de

The Munich Quantum Toolkit (MQT) is developed by the Chair for Design Automation at the Technical University of Munich and supported by the Munich Quantum Software Company (MQSC). Among others, it is part of the Munich Quantum Software Stack (MQSS) ecosystem, which is being developed as part of the Munich Quantum Valley (MQV) initiative.

Munich Quantum Valley
MQSS Munich Quantum Software Stack
MUNICH QUANTUM SOFTWARE COMPANY

Chair for Design Automation
TUM School of Computation, Information and Technology
Technical University of Munich

Leave a ★

MUNICH QUANTUM TOOLKIT

1 MQT ProblemSolver

Application

A Tool for Solving Problems using Quantum Computing

github.com/munich-quantum-toolkit/problemsolver

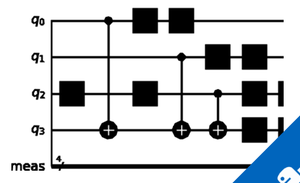
PyPI

2 MQT Bench

Application

A Quantum Circuit Benchmark Suite

github.com/munich-quantum-toolkit/bench



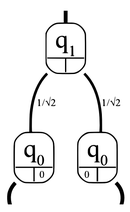
PyPI

3 MQT DDSIM

Simulation

A Tool for Classical Quantum Circuit Simulation based on Decision Diagrams

github.com/munich-quantum-toolkit/ddsim



PyPI

4 MQT YAQS

Simulation

A Tool for Simulating Open Quantum Systems, Noisy Quantum Circuits and Realistic Quantum Hardware

github.com/munich-quantum-toolkit/yaqs

Yet Another YAQS Quantum Simulator

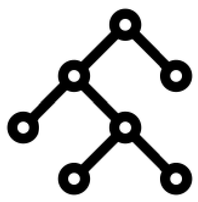
PyPI

5 MQT Predictor

Compilation

A Tool for Determining Good Quantum Circuit Compilation Options

github.com/munich-quantum-toolkit/predictor



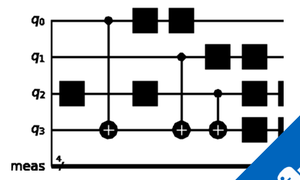
PyPI

6 MQT SyReC

Compilation

A Tool for the Synthesis of Reversible Circuits/Quantum Computing Oracles

github.com/munich-quantum-toolkit/syrec



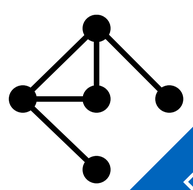
PyPI

7 MQT QMAP

Compilation

A Tool for Quantum Circuit Mapping and Clifford Circuit Optimization/Synthesis

github.com/munich-quantum-toolkit/qmap



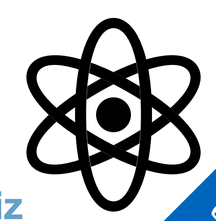
PyPI

8 MQT NAViz

Compilation

An Application to Visualize Compilation Output for Neutral Atom Quantum Computers

github.com/munich-quantum-toolkit/naviz



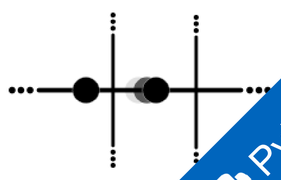
PyPI

9 MQT IonShuttler

Compilation

A Tool for Generating Shuttling Schedules for QCCD Architectures

github.com/munich-quantum-toolkit/ionshuttler



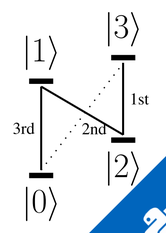
PyPI

10 MQT Qudits

Compilation

A Tool for Compiling to High-Dimensional Quantum Systems

github.com/munich-quantum-toolkit/qudits



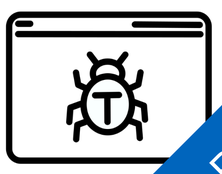
PyPI

11 MQT Debugger

Verification

A Tool for Debugging Quantum Circuits which can be integrated into your IDE

github.com/munich-quantum-toolkit/debugger



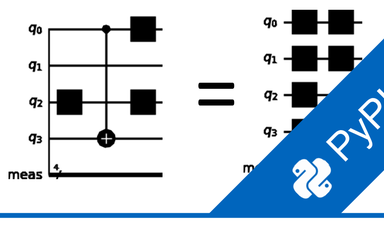
PyPI

12 MQT QCEC

Verification

A Tool for Quantum Circuit Equivalence Checking

github.com/munich-quantum-toolkit/qcec



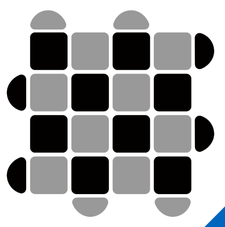
PyPI

13 MQT QECC

QEC

A Tool for Quantum Error Correcting Codes

github.com/munich-quantum-toolkit/qecc



PyPI

14 MQT DDVis

Data Structures

A Web-Application Visualizing Decision Diagrams for Quantum Computing

github.com/munich-quantum-toolkit/ddvis

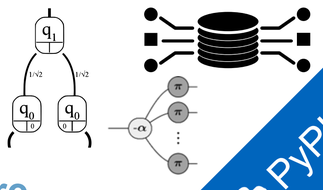
PyPI

15 MQT Core

Data Structures

The Backbone of the MQT Intermediate Representation (IR) Decision Diagram and ZX Package

github.com/munich-quantum-toolkit/core



PyPI

16 MQT QuSAT

Core Methods

A Tool for Encoding Quantum Computing using Satisfiability Testing (SAT) Techniques

$F \wedge (x_1 \wedge \neg x_2)$
$F \wedge (x_3 \wedge x_2)$

github.com/munich-quantum-toolkit/qusat

PyPI

Technische Universität München

European Research Council
Established by the European Commission

Munich Quantum Valley