Introduction to the Munich Quantum Toolkit (MQT)

Dingchao Gao

Institute of Software Chinese Academy of Sciences

June 25, 2024

Overview of MQT

- **What is MQT?**
  - developed by the Chair for Design Automation at the Technical University of Munich
  - provides advanced design automation methods and software tools for quantum computing
  - their objective is to provide solutions for design tasks across the entire quantum software stack

Figure: MQT includes high-level support for end users in realizing their applications, efficient methods for the classical simulation, compilation, and verification of quantum circuits, tools for quantum error correction, support for physical design, and more.

Key Features

- In all these tools, It try to utilize data structures (such as decision diagrams or the ZX-calculus) and core methods (such as reasoning engines) to facilitate the efficient handling of quantum computations
- Highlight major features of MQT.
  - Quantum Circuit Simulation
  - Quantum Circuit Compilation
  - Error Correction and Mitigation
- Unique Selling Points
  - How MQT stands out from other similar toolkits.
  - Any unique algorithms or methods used.

Modules and Components

- Detailed description of each major module or component.
  - Simulation Module
  - Compilation Module
  - Error Correction Module
- How these modules interact with each other.
- Examples of use cases for each module.

Installation and Setup

- Brief guide on how to install MQT.
  - Supported platforms (Windows, macOS, Linux).
  - Dependencies and prerequisites.
- Initial setup and configuration.

Usage and Examples

- Basic usage examples.
  - Simple quantum circuit simulation.
  - Compiling a quantum algorithm.
  - Applying error correction techniques.
- Advanced usage scenarios.
  - Integrating MQT into larger quantum computing projects.
  - Customizing MQT for specific research needs.

Community and Support

- Community engagement.
  - Online forums, discussion groups, and social media presence.
  - How to contribute to the development of MQT.
- Documentation and Tutorials.
  - Overview of available documentation and learning resources.
  - Official tutorials and example projects.

Future Developments

- Upcoming features and improvements.
- Long-term vision and goals for MQT.
- Opportunities for collaboration and contribution.

Conclusion

- Recap of the key points covered in the presentation.
- Encourage the audience to explore and use MQT.
- Provide contact information or resources for further inquiries.

Q&A Session

- Open the floor for questions from the audience.
- Prepare to answer common questions about MQT's capabilities, usage, and future plans.