CATEGORY: **DEVELOPER - TOOLS & LIBRARIES - DT06**
POSTER **P5317**
CONTACT NAME **Ying-Jer Kao: yjkao@phys.ntu.edu.tw**

GPU TECHNOLOGY CONFERENCE

Uni10-The Universal Tensor Network Library

An Open-Source Library for Tensor Network Algorithms

http://uni10.org

Department of Physics, National Taiwan University
Yun-Da Hsieh, Ying-Jer Kao
<yjkao@phys.ntu.edu.tw>

What is Uni10?

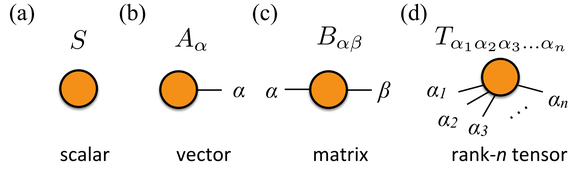
Uni10 is a general purpose tensor network library. It has comprehensive tensor operations and network construction methods. With Nvidia GPUs, applications using Uni10 can achieve 50x speedup by simply linking to the GPU version of the library without changing the code.

Uni10 distinguishes itself from existing tools with the following features:

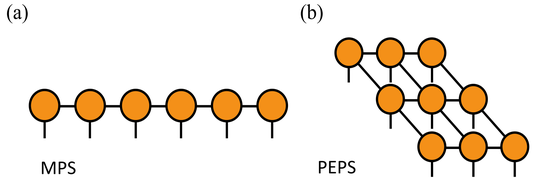
- *Fully implemented in C++.*
- *Aimed toward applications in tensor network algorithms.*
- *Provides basic tensor operations with an easy-to-use interface.*
- *Provides a class* **Network** *to process and store the graphical representations of the networks.*
- *Provides an engine to construct and analyze the contraction tree for a given network.*
- *Provides a collection of Python wrappers called* **pyUni10** *which interact with the compiled C++ library to take advantage of the Python language for better code readability and faster prototyping, without the sacrifice of speed.*
- *Provides behind-the-scene optimization and acceleration.*

Tensor Network Tensors are widely used in different branches of sciences and engineering, such as quantum physics and quantum chemistry, pattern and image recognition, signal processing, machine learning and computational neuroscience to represent very complicated multidimensional data with multiple aspects. Tensor networks can be used as a low-dimension and low-rank approximation of these complicated data through tensor decomposition.

Representing Tensor Networks Tensors and tensor networks can be represented graphically.



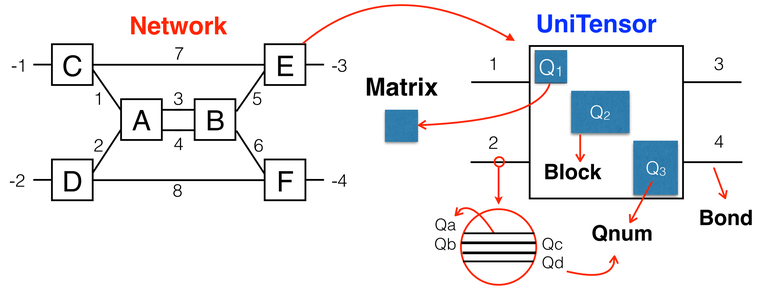
Internal lines are contracted, and external lines represent the open indices.



Inside Uni10

A tensor is represented in Uni10 as a **UniTensor** with the following components:

- **Bond** An index of a tensor, possessing dimension and symmetry quantum numbers (Qnum: U1 and Z2).
- **Element** Elements of tensor. With symmetry, elements are in a block-diagonal form.
- **Label** Labeling the bonds to be used for tensor operations



A **Network** holds the information of how tensors are connected.

CUDA implementation

The Uni10 implementation on Nvidia© GPU with CUDA uses the following strategies:

Memory managements

- Elements are allocated in GPU device memory if possible to avoid data transfer between device and host.
- For large tensors, elements are allocated in the main memory and transferred piece-wise to GPU for linear algebra operations, *e.g.* tensor contraction.
- Automatic transfer when needed, *e.g.* printing tensor.

Operations on GPU

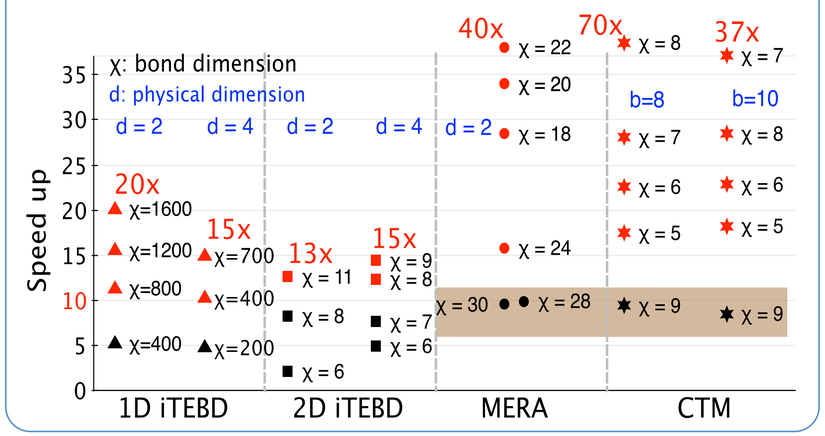
- Operations with CUDA: allocation, memory copy, resize(), permute(), and etc.
- Linear algebra with cuBLAS and CULA: contraction, svd(), eigh(), and etc.

Unified API

- The API is designed so that the GPU is almost invisible to the end user, with the options for simple memory managements.
- The same CPU code can start utilizing GPU acceleration by linking to a GPU version of Uni10.

Benchmarks

We benchmarked several tensor network algorithms used in quantum physics with Uni10 accelerated by Nvidia Tesla M2090 with 6GB device memory. Results are compared against a single Intel(R) Core(TM) i7-3770K CPU @ 3.50GHz. In general, larger tensor sizes can achieve better speedups before tensors are too large to fit entirely on the device memory.



Conclusions and Outlook

- Uni10 implements a concise and intuitive label-based operation API for tensor-network algorithm.
- Using Uni10, users can painlessly integrate their code the GPU to get Significant speedup, without any modifications of the code.
- Uni10 also provides a Python wrapper **pyUni10** which allows users to easily utilize these features in python.
- Matlab wrapper for Uni10 will be released soon.
- Multi-GPU support will be added to resolve the issues with large tensors.

References

[1] Y.-D. Hsieh, P. Chen, and Y.-J. Kao, *Uni10: an open-source library for tensor network algorithms* (Preprint).

[2] R. Orus, A Practical Introduction to Tensor Networks: Matrix Product States and Projected Entangled Pair States, Annals of Physics 349,117 (2014).

[3] A. Cichocki, *Era of Big Data Processing: A New Approach via Tensor Networks and Tensor Decompositions* (Preprint arXiv:1403.2048)

Acknowledgements

This work is supported by MOST of Taiwan through grant number 102-2112-M-002-003-MY3.