Python Package for Universal Tensor-Networks Simple-Update Simulations

Roy Elkabetz[1, *]

[1] *Department of Computer Science, Technion, 3200003 Haifa, Israel*

(Dated: August 1, 2022)

`tnsu` is a Python package for universal Tensor-Networks (TN) Simple-Update (SU) simulations. Simple Update is a computationally efficient TN algorithm for finding ground-states TN representations of local Hamiltonians. The algorithm calculates the dynamics of many-body spin systems that sit on an N-dimensional lattice using *real* and *imaginary time evolution* methods. The `tnsu` package also supports the trivial-SU algorithm for finding TN canonical representations of many-body quantum-states, which for TN with tree-like topology is directly related to their partitioning using the Schmidt Decomposition. The code is publicly available on PyPI and in the corresponding GitHub repository `https://github.com/RoyElkabetz/Tensor-Networks-Simple-Update`.

INTRODUCTION

Tensor Networks (TN) is a mathematical framework that uses a graphical representation of systems with a large number of degrees of freedom in order to understand their local structure and the relations between their parts. In the last decade, tensor networks were mostly used in physics-based fields such as quantum physics, statistical mechanics, and computational physics. The TN objective is to enable an efficient description of many-body quantum states using a network of inter-connected tensors, where a key component within the framework is the ability to approximate ground-states of many-body quantum spin systems and their local expectation values. There are a handful of algorithms under the TN umbrella that do just that, all of them lay on the continuum between accuracy and efficiency. In this paper, we present an open-source Python package named `tnsu`, for implementing the computationally efficient TN algorithm also known as Simple-Update (SU) and its variant trivial-Simple-Update (tSU), which as far as we know do not currently exist in an open-source setting.

METHOD

Simple Update

The `tnsu` Python package (an abbreviation for tensor-networks-simple-update) is an open-source simulation tool for finding approximate Tensor Network (TN) representations of ground states of spin lattices with local Hamiltonians. The tool is an implementation of the tensor network Simple-Update (SU) algorithm as described in Ref. [1] and illustrated in Fig.(1). The SU algorithm is a commonly used algorithm in the tensor networks community and it is well-known for its high efficiency although it might give inaccurate results in some cases, in particular in highly frustrated and/or critical systems. Nevertheless, also in such challenging cases, it is often used as an initial step for finding a crude approximation to the ground state, and later using more accurate, but also more computationally demanding methods such as the Cluster Update [2], the Full-Update [3] and others.

Trivial Simple Update

In addition to SU simulations, the `tnsu` package also supports the implementation of the trivial-Simple-Update (tSU) tensor network algorithm for fining canonical tensor network representations for networks with tree-like topology such as Matrix Product States (MPS) (with open boundary conditions), and approximated canonical representations for tensor networks with loops (i.e Projected Entangled Pair States (PEPS)) as it is described in Ref. [4] on the duality between the tSU algorithm to the well known Belief Propagation (BP) algorithm from the field of Probabilistic Graphical Models (PGM).

CONCLUSION

Hopefully, this tool would be beneficial to the Tensor Networks community and would encourage others to share their code as well. It is important to note that the package is still under development, so it might have bugs. I also would like to encourage any keen tensor network researcher to take part in the collaboration of further development of the `tnsu` package.

* elkabetzroy@gmail.com

[1] S. S. Jahromi and R. Orús, Physical Review B **99**, 195105 (2019).

[2] M. Lubasch, J. I. Cirac, and M.-C. Banuls, New Journal of Physics **16**, 033014 (2014).

[3] R. Orús, Annals of physics **349**, 117 (2014).

[4] R. Alkabetz and I. Arad, Physical Review Research **3**, 023073 (2021).

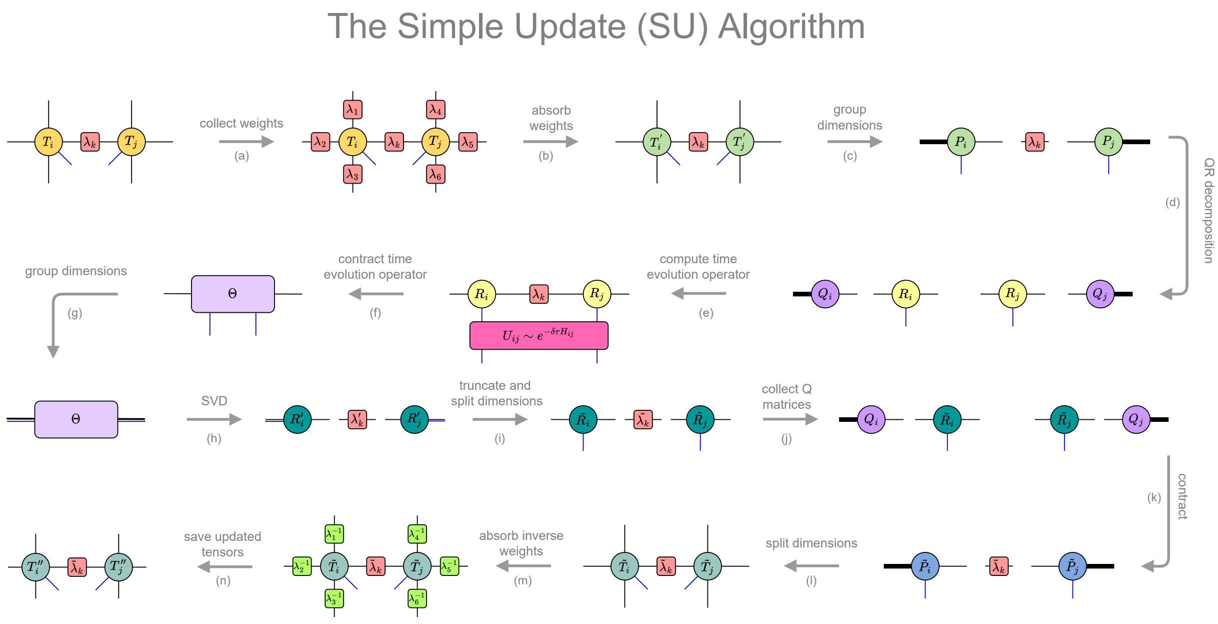

Figure 1. A step-by-step Simple Update algorithm illustration as it is implemented in `tnsu`. The trivial-SU implementation is identical to the SU one with the difference of $\delta\tau = 0$ (in step (e)). The blue and black legs of each tensor corresponds to its physical and virtual degrees of freedom respectively.