Computer Physics Communications 185 (2014) 1570–1581

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

The density matrix renormalization group algorithm on kilo-processor architectures: Implementation and trade-offs

Csaba Nemes [a,*], Gergely Barcza [c], Zoltán Nagy [a,b], Örs Legeza [c], Péter Szolgay [a,b]

[a] Faculty of Information Technology, Péter Pázmány Catholic University, Budapest, Hungary

[b] Cellular Sensory and Wave Computing Laboratory, Computer and Research Automation Institute, Hungarian Academy of Sciences, Budapest, Hungary

[c] Strongly Correlated Systems "Lendület" Research Group, Department of Theoretical Solid State Physics, Wigner Research Centre for Physics, Hungarian Academy of Sciences, Budapest, Hungary

ARTICLE INFO

Article history:
Received 21 September 2013
Received in revised form
10 February 2014
Accepted 18 February 2014
Available online 26 February 2014

Keywords:
Strongly correlated systems
DMRG
GPU acceleration
FPGA acceleration

ABSTRACT

In the numerical analysis of strongly correlated quantum lattice models one of the leading algorithms developed to balance the size of the effective Hilbert space and the accuracy of the simulation is the density matrix renormalization group (DMRG) algorithm, in which the run-time is dominated by the iterative diagonalization of the Hamilton operator. As the most time-dominant step of the diagonalization can be expressed as a list of dense matrix operations, the DMRG is an appealing candidate to fully utilize the computing power residing in novel kilo-processor architectures.

In the paper a smart hybrid CPU–GPU implementation is presented, which exploits the power of both CPU and GPU and tolerates problems exceeding the GPU memory size. Furthermore, a new CUDA kernel has been designed for asymmetric matrix–vector multiplication to accelerate the rest of the diagonalization. Besides the evaluation of the GPU implementation, the practical limits of an FPGA implementation are also discussed.

1. Introduction and related works

DMRG is a variational numerical approach developed to treat low-dimensional interacting many-body quantum systems efficiently [1–3]. In fact, it has become an exceptionally successful method to study the low energy physics of strongly correlated quantum systems which exhibit chain-like entanglement structure [4]. For example, it can be applied to simulate properties of anisotropic materials, such as polymers [5], or to describe accurately the electronic structure of open *d* shell molecules [6], which is beyond the capability of standard quantum chemical approaches. Additionally, the interacting system of atoms trapped in an optical lattice, proposed as physical implementation of quantum computer, is also tractable via DMRG [7].

Over the past ten years, the DMRG method has been also reformulated in terms of Matrix Product States (MPS) [8] leading to various extensions [9–11] and has been shown to be a special case of a more general set of methods, the so-called Tensor Network States (TNS) [12–14]. A common feature of all these algorithms is that computational tasks can be massively parallelized.

The original DMRG algorithm [1] was introduced in 1992 by Steven R. White and was formulated as a single threaded algorithm. In the past various works have been carried out to accelerate the DMRG algorithm on shared [15,16] and distributed memory [17–20] architectures, however, none of them took advantage of recent kilo-processor architectures: graphical processing unit (GPU) and field-programmable gate array (FPGA).

One of the first parallelizations was [15] converting the projection operation to matrix–matrix multiplications and accelerating them via OpenMP interface. In [19] a similar approach was presented for distributed memory environment (up to 1024 cores) optimizing the communication between the cores, while in [20] the acceleration of the computation of correlation function had been investigated. Recently, [16] presented an acceleration on shared memory architectures exploiting SU(2) symmetries, while [21] proposed a novel direction for parallelization via a modification of the original serial DMRG algorithm.

Graphical processing unit has been successfully employed in neighboring research areas to accelerate matrix operations. In [22] GPU is used to accelerate tensor contractions in Plaquette Renormalization States (PRS), which can be regarded as an alternative

* Corresponding author.
E-mail addresses: csnemes@gmail.com, nemes.csaba@itk.ppke.hu (C. Nemes).

http://dx.doi.org/10.1016/j.cpc.2014.02.021
0010-4655/

technique to Tensor Product States (TPS) or the DMRG algorithm. In [23] the second-order spectral projection (SP2) algorithm has been accelerated, which is an alternative technique to calculate the density matrix via a recursive series of generalized matrix–matrix multiplications.

In this paper we present the first attempt (to our best knowledge) to investigate how the DMRG method can utilize the enormous computing capabilities of novel kilo-processor architectures (GPU, FPGA). In case of GPU a smart hybrid CPU–GPU acceleration is presented, which tolerates problems exceeding the GPU memory size, consequently, supporting wide range of problems and GPU configurations. Contrary to the previous acceleration attempts not only the projection operation is accelerated, but further parts of the diagonalization are also computed on the GPU. In case of FPGA the performance limits of a possible implementation are estimated and discussed.

The rest of the paper is organized as follows. Section 2 describes the models which are used as test cases to demonstrate the operation of the algorithm. Symmetries which can be exploited to decrease the computational requirements of the algorithm and the algorithm itself are presented in Sections 3 and 4, respectively. Acceleration on GPU is presented in three Sections 5–7, while limits of an FPGA implementation are described in Section 8. Finally, implementation results and conclusions are given in Sections 9 and 10, respectively.

2. Investigated models

In order to illustrate the underlying features of the algorithm it is applied to the so-called spin-$\frac{1}{2}$ Heisenberg model and the spin-$\frac{1}{2}$ Hubbard model. The selected models describe how to compute the Hamiltonian of the system of interest, while the main task is to find some of the low-lying eigenvalues and eigenvectors of the Hamiltonian by a diagonalization algorithm. In practice instead of solving the problem for the complete Hilbert space directly, various physical phenomena can be exploited to reduce the complexity of the problem.

2.1. Heisenberg model

The Heisenberg model describes the physics of magnetic systems and provides theoretical description of various experimental measurements. In the model a magnetic system is simulated on a lattice of interacting *spins*. A microscopic magnetic moment (spin) is localized at each lattice site j and described by a quantized, two-valued variable, $\sigma_j \in \{\uparrow, \downarrow\}$, related to the two possible orientations of the spin. Limiting the interactions to only neighboring spins – which is often a good approximation – the Hamiltonian of the model is written as

$$H = \frac{1}{2}\sum_{j=1}^{N-1}\left(S_j^+ S_{j+1}^- + S_j^- S_{j+1}^+\right) + \Delta\sum_{j=1}^{N-1} S_j^z S_{j+1}^z \tag{1}$$

where S_j^+, S_j^- operators change, while S_j^z measures the orientation of the spin on lattice site j. The overall behavior of the system can be tuned via the relevant parameter Δ. The explicit matrix representation of an operator $\mathcal{O}_j$ acting on site j of a chain with N spins is given as

$$\mathcal{O}_j = \bigotimes_{i=1}^{j-1}\mathbb{I}\otimes\mathcal{O}\otimes\bigotimes_{i=j+1}^{N}\mathbb{I} \tag{2}$$

where $\mathbb{I}$ is the identity and $\mathcal{O}$ is one of the followings

$$S^+ = \begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}, \qquad S^- = \begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix}, \qquad S^z = \frac{1}{2}\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}. \tag{3}$$

The Hamiltonian of N spins acts on the tensor product space of dimension 2^N, that is the dimension of the complete Hilbert space grows exponentially as the size of the system increases. In the rest of the paper benchmark results are shown for $\Delta = 1$.

2.2. Hubbard model

The Hubbard model was introduced to describe electrons in solids to characterize the transition between insulating and conducting systems. The single-band Hubbard model provides appropriate description of low temperature systems where all particles are in the lowest Bloch band and the long-ranged interactions between the particles can be neglected due to strong screening effects [24]. More recently various multi-band Hubbard models have been applied to high-temperature superconductivity [25] and systems of higher spin to understand the behavior of optically trapped ultracold atoms [7].

In the general spin-F system each lattice site is characterized by $2F+1$ two dimensional vectors. Each vector is assigned with a distinct label (from $\{-F, -F+1, \ldots, F-1, F\}$) called spin polarization value (denoted by σ). A vector assigned to a spin polarization σ describes two orthogonal states: the site is occupied ($[0; 1]$) by the particle of spin polarization σ or not ($[1; 0]$). As a consequence, a lattice site of spin-F possesses 2^{2F+1} internal degrees of freedom.

The lattice model of interacting particles of spin-F consists of two competing terms: the kinetic term, which describes the tunneling of particles between neighboring lattice sites, and the local potential term, which describes on-site density–density interaction measuring the attraction or repulsion between the interacting particles. The single-band, fermionic Hubbard model of spin-F is defined on a chain with N sites as

$$H = -t\sum_{j=1}^{N-1}\sum_{\sigma=-F}^{F}\left(c_{j,\sigma}^\dagger c_{j+1,\sigma}^\dagger + \text{h.c.}\right) + \frac{U}{2}\sum_{j=1}^{N}\sum_{\sigma\neq\sigma'} n_{j,\sigma}^\dagger n_{j,\sigma'}^\dagger \tag{4}$$

where t measures the hopping amplitude between neighboring sites and U is the interaction strength. Creation and annihilation operator acting on site j with spin polarization σ, denoted as $c_{j,\sigma}^\dagger$ and $c_{j,\sigma}$, adds or removes a particle located on site j with spin polarization σ. The particle density of spin polarization σ on site j is measured by operator $n_{j,\sigma}^\dagger = c_{j,\sigma}^\dagger c_{j,\sigma}^\dagger$. The explicit matrix representation of an operator $\mathcal{O}_{j,\sigma}$ acting on site j and polarization σ is constructed as

$$\mathcal{O}_{j,\sigma} = \bigotimes_{i=1}^{F'(j-1)}\Phi\otimes\mathcal{O}_\sigma\otimes\bigotimes_{i=F'(j+1)}^{F'N}\mathbb{I} \tag{5}$$

$$\mathcal{O}_\sigma = \bigotimes_{i=-F}^{\sigma-1}\Phi\otimes\mathcal{O}\otimes\bigotimes_{i=\sigma+1}^{F}\mathbb{I} \tag{6}$$

$$\Phi = \begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix} \tag{7}$$

where $F' = 2F+1$, $\mathbb{I}$ is the identity, Φ is the fermionic phase-factor and $\mathcal{O}$ is one of the followings

$$c^\dagger = \begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix}, \qquad c = \begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}. \tag{8}$$

The Hamiltonian describing the spin-F system of N lattice sites acts on the tensor product space of dimension $2^{F'N}$, and similarly to the Heisenberg model, the dimension of the complete Hilbert space blows up exponentially. Comparing to the bosonic operators of the Heisenberg model, the key differences in the construction of operators are the appearance of internal quantum number, σ, and the presence of the phase-factor describing the antisymmetric nature