

Max-Planck-Institut  
für Mathematik  
in den Naturwissenschaften  
Leipzig

Computation of the Hartree-Fock Exchange by  
the Tensor-structured Methods

(revised version: March 2010)

by

*Venera Khoromskaia*

Preprint no.: 25

2009





# Computation of the Hartree-Fock Exchange by the Tensor-structured Methods

Venera Khoromskaia  
Max-Planck-Institute for Mathematics in the Sciences,  
Inselstr. 22-26, D-04103 Leipzig, Germany.  
{vekh}@mis.mpg.de

## Abstract

We introduce the novel numerical method for fast and accurate evaluation of the exchange part of the Fock operator in the Hartree-Fock equation which is the (non-local) integral operator in  $\mathbb{R}^3 \times \mathbb{R}^3$ . Usually, this challenging computational problem is solved by laborious analytical evaluation of the two-electron integrals using “analytically separable” Galerkin basis functions, like Gaussians. Instead, we employ the agglomerated “grey-box” numerical computation of the corresponding six-dimensional integrals in the tensor-structured format which does not require analytical separability of the basis set. The core of our method is the low-rank tensor representation of arising functions and operators on  $n \times n \times n$  Cartesian grid, and implementation of the corresponding multi-linear algebraic operations in the tensor product format. Linear scaling of the tensor operations, including the 3D convolution product, with respect to the one-dimension grid size  $n$  enables computations on huge 3D Cartesian grids thus providing the required high accuracy. The presented algorithm for computation of the exchange operator and a recent tensor method of the Coulomb matrix evaluation are the main building blocks in the numerical solution of the Hartree-Fock equation by the tensor-structured methods. These methods provide the new tool for algebraic optimization of the Galerkin basis in the case of large molecules.

*AMS Subject Classification:* 65F30, 65F50, 65N35, 65F10

*Key words:* Hartree-Fock operator, exchange matrix, canonical model, discrete tensor convolution, tensor-structured methods, tensor-product basis functions.

## 1 Introduction

In recent decades much progress has been achieved in the development of the canonical and Tucker-type decomposition algorithms in application to the problems of independent component analysis, signal processing and higher order statistics, see [3, 4] and a comprehensive survey on tensor decomposition methods [18].

Theoretical analysis of the multilinear tensor product approaches for the treatment of some multivariate operators and functions arising in scientific computing have been addressed in [5, 6, 8, 12]. Application of the tensor decomposition algorithms to discretized

multivariate functions and operators [10, 14, 13, 11] showed that methods of the multi-way analysis can be applied to the numerical solution of basic equations of mathematical physics having strong requirements on accuracy of the results. In particular, the Tucker and canonical tensor product approximations allow to reduce dramatically the complexity of accurate function and operator calculus in  $\mathbb{R}^d$ ,  $d \geq 3$ , realized on large Cartesian grids [15]. Tensor-structured algorithms, working as the “grey-box” schemes, appear to be efficient in electronic structure calculations [11, 15, 16, 17]. In fact, they provide the required high accuracy of computations, and also give an opportunity to avoid obligatory analytically treatable basis sets, like Gaussians, for the solution of the Hartree-Fock equation.

Here, we develop the grid-based tensor-structured method for the computation of the Hartree-Fock exchange represented on  $n \times n \times n$  Cartesian grid. Numerical complexity of the respective algorithm scales linearly in the univariate grid size  $n$ ,  $O(n)$ . Moreover, this method is free from the conventional limitations on the basis functions, and can be useful for further algebraic optimization of the Galerkin basis in the case of large molecules.

The Hartree-Fock model provides a mean-field approximation for the ground state of many-electron systems. This implies the solution of a nonlinear eigenvalue problem in  $\mathbb{R}^3$

$$\left(-\frac{1}{2}\Delta + V_{nuc} + V_H - V_x\right) \varphi_a(x) = \lambda_a \varphi_a(x), \quad x \in \mathbb{R}^3, \quad (1.1)$$

for the  $N_{orb}$  lowest eigenvalues  $\lambda_a$  and spatial eigenfunctions  $\varphi_a$  ( $a = 1, \dots, N_{orb}$ ), in the case of a closed-shell  $N$  electron system,  $N = 2N_{orb}$ . Equation (1.1) corresponds to a nonlinear single-particle Schrödinger equation in  $\mathbb{R}^3$ , where the potentials  $V_H$  and  $V_x$  represent a mean-field acting on a single electron which is generated by the remaining  $N - 1$  electrons in the system. Here, an external potential  $V_{nuc}$  contains the bare Coulomb- or pseudopotentials of the nuclei.

The tensor-structured methods developed in [14, 10, 13] have been successfully applied for the highly accurate *grid-based* numerical computation of the Hartree potential and the Coulomb matrix in the Hartree-Fock equation [15, 17]. For the efficient computation of the Hartree potential in (1.1),

$$V_H(x) := \int_{\mathbb{R}^3} \frac{\rho(y)}{\|x - y\|} dy, \quad x \in \mathbb{R}^3, \quad (1.2)$$

which corresponds to the convolution of the Coulomb potential with the electron density

$$\rho(y) = 2 \sum_{a=1}^{N_{orb}} \varphi_a(y) \varphi_a^*(y), \quad (1.3)$$

we used the low-rank tensor product representation of the electron density  $\rho$  and the convolving kernel on  $n \times n \times n$  Cartesian grid and performed multilinear operations in the tensor-product format.

In the present paper, we consider the tensor product approximation of the nonlocal (integral) exchange operator  $V_x$  in the Hartree-Fock equation. Note that calculation of the exchange Galerkin matrix in the Hartree-Fock equation is a challenging problem due to the nonlocal character of the exchange operator

$$(V_x \psi)(x) := \sum_{b=1}^{N_{orb}} \int_{\mathbb{R}^3} \frac{\varphi_b(x) \varphi_b^*(y)}{\|x - y\|} \psi(y) dy, \quad x \in \mathbb{R}^3, \quad (1.4)$$

leading to the integration in six dimensions (see (3.1)). This problem is usually solved analytically by evaluating the so-called two-electron integrals using separable basis sets like Gaussians, see [27, 20] and references therein.

Here, we propose and implement the agglomerated grid-based computation of the Hartree-Fock exchange (1.4). We apply the tensor product approximation of arising operators and functions on  $n \times n \times n$  Cartesian grid and use multilinear tensor operations providing linear scaling with respect to the one-dimension grid size  $n$ ,  $O(n)^1$ . We use the fast tensor product convolution for the multivariate functions in  $\mathbb{R}^d$ ,  $d \geq 3$ , already employed in [15] for evaluating  $V_H$ , which provides the complexity  $O(dn \log n)$ , in our case  $d = 3$ . The tensor product convolution developed in [13] considerably outperforms the benchmark algorithm based on the 3D Fast Fourier Transform (FFT) having the cost  $O(n^3 \log n)$ . Note, that in our numerical examples, we use equal sizes  $n$  of the 3D Cartesian grid for three spatial dimensions and do not use the information on symmetry of the molecules. As the input data we only use the discrete representation of the Galerkin basis functions, therefore, we may say, this is a “grey-box” algorithm.

Our initial algorithm for evaluation of (1.4) has the complexity  $O(n \log n R_0^2 + n_{ef} R_0^4 N_{orb})$ , where  $n_{ef} \ll n$  is the “effective” univariate grid size, and  $R_0$  is the number of Galerkin basis functions. Here we reduce the constant in the linear complexity scaling in  $n$  by truncating the regions of the computation intervals, where the values of rapidly decaying basis functions (in particular, Gaussians) are less than a threshold controlling the accuracy of computations. Thus, we have for the number of grid points in effective support of the interacting vectors,  $n_{ef} = \alpha n$ , with  $\alpha$  much less than 1.

To reduce the  $R_0$ -asymptotics to  $O(R_0^3)$ , we further apply the canonical-to-Tucker algorithm for decreasing the ranks of intermediate results after every convolution step. The corresponding rank reduction algorithms are considered in [15].

The main advantage of the proposed computational scheme is the ability to avoid the “analytically separable” rank-1 basis sets like Gaussians, which are obligatory for the standard approaches. It is well known that the size of the Gaussian basis sets grows essentially for larger molecules, which makes the related Hartree-Fock problem with the complexity scaling as  $R_0^3$  computationally unfeasible. Here, we use the discretized Gaussians mostly for the sake of convenient comparison of the accuracy of computations with the benchmark results of the standard MOLPRO package [26]. Indeed, we can employ as the Galerkin basis any appropriate set of functions which are separable algebraically (say, using the Tucker decomposition), with the ranks larger or equal to 1 and complying with the approximation requirements. Therefore, the tensor-structured method proposed in this paper provides new means for the algebraic optimization of the Galerkin basis in the case of large molecules.

The accuracy of the computation on a particular grid is estimated by  $O(h^2)$ , where  $h = O(n^{-1})$  is the step-size of the grid. We achieve  $O(h^3)$  accuracy in our evaluation of the exchange matrix by using the Richardson extrapolation on a couple of consequent grids. Usually, the univariate size of the computational box for small to medium size molecules is in the range of  $14 \div 20 \text{ \AA}$ . Since the tensor-structured techniques enable computations on huge 3D Cartesian grids, the univariate step-sizes of applied grids are in a range from

---

<sup>1</sup>Note that commonly used notation as “linear in the problem size” for the problems in three spatial dimensions often means linear complexity with respect to the volume size which is  $V = n^3$ .

$h \approx 2 \cdot 10^{-2} \text{ \AA}$  for  $n = 1024$ , up to  $h \approx 8 \cdot 10^{-4} \text{ \AA}$  for the benchmark grids with the number of entries  $n^3 = 16384^3$ .

The rest of the paper is organized as follows. In Section 2 we recall the definitions of the basic rank-structured formats and describe the multilinear tensor-product operations in the rank- $R$  canonical format. In Section 3, we discuss the representation of the exchange operator in the particular Galerkin basis and the discrete computational scheme. The latter does not depend on the character of the basis functions, and allows arbitrary vectors of the canonical agglomerated representation of a given 3D tensor. We give the detailed description of the algorithm and provide the complexity estimate. Section 4 describes numerical results of computations of the Hartree-Fock exchange matrix for the pseudopotential case of some organic molecules and all electron case of water molecule using huge 3D Cartesian grids up to size  $16384^3$ . Figures illustrate the accuracy  $O(h^3)$  and the linear scaling of computation time in the univariate grid size  $n$ . Numerical experiments are performed in Matlab 7.6, on a standard SUN station. The results of computations are given in comparison with the output of the standard benchmark package MOLPRO [26].

Tensor-structured computation of the Hartree-Fock exchange, along with the tensor-based algorithms for calculating the Coulomb matrix considered in [15] are the main building blocks in the recent grid-based numerical solution of the Hartree-Fock equation by the tensor-structured methods (see a 3D nonlinear EVP solver [16]).

## 2 Tensor-structured representation of multivariate functions and operators

### 2.1 Rank-structured tensor approximation

A tensor of order  $d$  is a multidimensional array of real/complex data whose elements are referred by using a tensor-product index set  $\mathcal{I} = I_1 \times \dots \times I_d$ . We use the common notation

$$A = [a_{i_1, \dots, i_d} : i_\ell \in I_\ell] \in \mathbb{R}^{\mathcal{I}}, \quad I_\ell = \{1, \dots, n_\ell\}, \quad \ell = 1, \dots, d,$$

to denote a  $d$ th order tensor, and  $\mathbf{n}$  for the  $d$ -tuple  $(n_1, \dots, n_d)$ . A tensor  $A$  is an element of the tensor-product linear space  $\mathbb{V}_{\mathbf{n}} = \otimes_{\ell=1}^d \mathbb{V}_\ell$  with  $\mathbb{V}_\ell = \mathbb{R}^{I_\ell}$ , equipped with the Euclidean scalar product  $\langle \cdot, \cdot \rangle : \mathbb{V}_{\mathbf{n}} \times \mathbb{V}_{\mathbf{n}} \rightarrow \mathbb{R}$ , defined as

$$\langle A, B \rangle := \sum_{(i_1, \dots, i_d) \in \mathcal{I}} a_{i_1, \dots, i_d} b_{i_1, \dots, i_d} \quad \text{for } A, B \in \mathbb{V}_{\mathbf{n}}. \quad (2.1)$$

Assume for simplicity that  $\dim \mathbb{V}_\ell = \#I_\ell = n$  for all  $\ell = 1, \dots, d$ , then the number of entries in  $V$  amounts to  $n^d$ , hence growing exponentially in  $d$ .

To get rid of exponential scaling in the dimension, the approximate data-sparse ‘‘rank structured’’ representations of tensors in  $\mathbb{V}_{\mathbf{n}}$  can be applied. As the simplest rank structured ansatz, we make use of rank-1 tensors. Specifically, the *tensor product* of vectors  $u_\ell = \{u_{\ell, i_\ell}\}_{i_\ell \in I_\ell} \in \mathbb{V}_\ell$  ( $\ell = 1, \dots, d$ ) forms the canonical rank-1 tensor

$$A \equiv [u_{\mathbf{i}}]_{\mathbf{i} \in \mathcal{I}} = u_1 \otimes \dots \otimes u_d \in \mathbb{V}_{\mathbf{n}} \quad \text{with entries} \quad u_{\mathbf{i}} = u_{1, i_1} \cdots u_{d, i_d},$$

which requires only  $dn$  numbers to store it.

We define a tensor in *the canonical format*

$$A_{(R)} = \sum_{k=1}^R c_k u_k^{(1)} \otimes \dots \otimes u_k^{(d)}, \quad c_k \in \mathbb{R}, \quad (2.2)$$

with normalised vectors  $u_k^{(\ell)} \in \mathbb{V}_\ell$  ( $\ell = 1, \dots, d$ ), where the minimal parameter  $R \in \mathbb{N}$  in (2.2) is called the rank (or canonical rank) of a tensor. In our tensor-structured computations, we use the rank- $R$  canonical representation for the multilinear operations.

Given the rank parameter  $\mathbf{r} = (r_1, \dots, r_d)$ , we can represent the initial tensor  $A$  in the so-called Tucker format

$$A \approx A_{(\mathbf{r})} = \sum_{\nu_1=1}^{r_1} \dots \sum_{\nu_d=1}^{r_d} \beta_{\nu_1, \dots, \nu_d} v_{\nu_1}^{(1)} \otimes \dots \otimes v_{\nu_d}^{(d)}, \quad (2.3)$$

with some vectors  $v_{\nu_\ell}^{(\ell)} \in \mathbb{V}_\ell = \mathbb{R}^{I_\ell}$  ( $1 \leq \nu_\ell \leq r_\ell$ ), which form the orthonormal basis of  $\text{span}\{v_\nu^{(\ell)}\}_{\nu=1}^{r_\ell}$  ( $\ell = 1, \dots, d$ ). Here we call the parameter

$$r = \max_{\ell} \{r_\ell\}$$

the *maximal Tucker rank*. For classes of function related tensors, the choice  $r = O(\log n)$  ensures the approximation order  $O(1/n)$  [5, 6, 10]. The coefficients tensor  $\beta = [\beta_{\nu_1, \dots, \nu_d}]$ , that is an element of a tensor space

$$\mathbb{B}_{\mathbf{r}} = \mathbb{R}^{J_1 \times \dots \times J_d}, \quad J_\ell = \{1, \dots, r_\ell\}, \quad \ell = 1, \dots, d, \quad (2.4)$$

is called the *core tensor*.

Introducing the (orthogonal) side matrices  $V^{(\ell)} = [v_1^{(\ell)} \dots v_{r_\ell}^{(\ell)}]$ , we then use a tensor-by-matrix contracted product to represent the Tucker decomposition of  $A_{(\mathbf{r})}$ ,

$$A_{(\mathbf{r})} = \beta \times_1 V^{(1)} \times_2 V^{(2)} \dots \times_d V^{(d)}. \quad (2.5)$$

In present computations we also use the mixed Tucker-canonical format,

$$A_{(\mathbf{r})} = \left( \sum_{k=1}^R b_k u_k^{(1)} \otimes \dots \otimes u_k^{(d)} \right) \times_1 V^{(1)} \times_2 V^{(2)} \times_3 \dots \times_d V^{(d)}.$$

that is visualized in Figure 2.1. In this case the Tucker core is represented by a rank- $R$  canonical tensor. More detailed description of the tensor decomposition algorithms and of the multigrid rank reduction scheme based on the canonical-to-Tucker approximation is given in [15].

## 2.2 Multilinear operations in the tensor product format

In our numerical scheme we apply the following linear operations with  $d$ th order tensors:

1. summation of tensors;

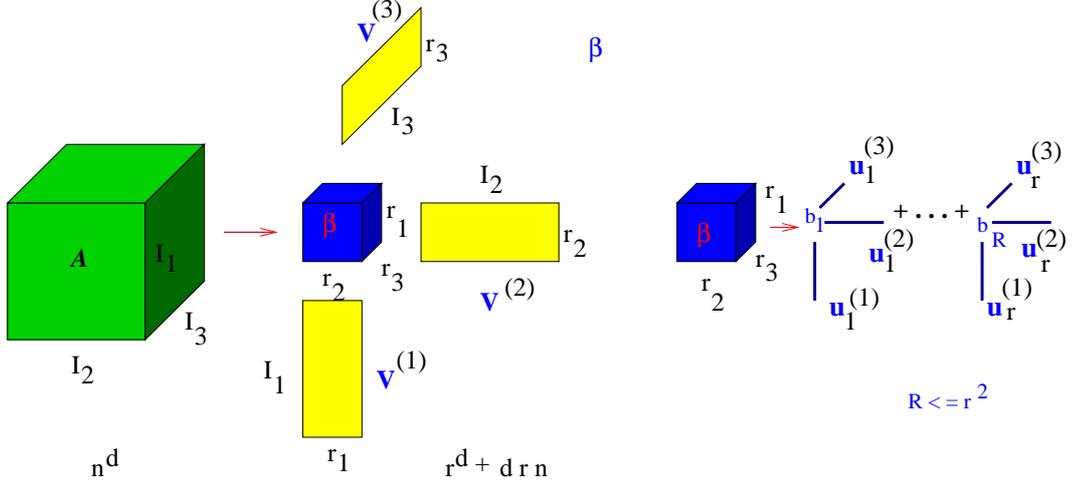


Figure 2.1: Mixed Tucker-canonical format.

2. scalar product of tensors;
3. Hadamard product of tensors;
4. convolution product of tensors.

A comprehensive description of the multi-linear tensor-product operations for the multi-dimensional tensors is presented in the survey [18], see also [14, 13, 22] for details on function related tensors.

Let us consider tensors  $A_1, A_2$ , represented in the rank- $R$  canonical format, (2.2),

$$A_1 = \sum_{k=1}^{R_1} c_k u_k^{(1)} \otimes \dots \otimes u_k^{(d)}, \quad A_2 = \sum_{m=1}^{R_2} b_m v_m^{(1)} \otimes \dots \otimes v_m^{(d)}, \quad (2.6)$$

with normalized vectors  $u_k^{(\ell)}, v_m^{(\ell)} \in \mathbb{R}^{I_\ell}$ . (For simplicity of notation, we consider  $n_\ell = n$ .)

1. A sum of two canonical tensors given by (2.6) can be written as

$$A_1 + A_2 = \sum_{k=1}^{R_1} c_k u_k^{(1)} \otimes \dots \otimes u_k^{(d)} + \sum_{m=1}^{R_2} b_m v_m^{(1)} \otimes \dots \otimes v_m^{(d)}, \quad (2.7)$$

resulting in the canonical tensor with the rank  $R_S = R_1 + R_2$ . This operation has no cost since it is simply a concatenation of two tensors.

2. For given canonical tensors  $A_1, A_2$ , the *scalar product* (2.1) can be computed by

$$\langle A_1, A_2 \rangle := \sum_{k=1}^{R_1} \sum_{m=1}^{R_2} c_k b_m \prod_{\ell=1}^d \langle u_k^{(\ell)}, v_m^{(\ell)} \rangle. \quad (2.8)$$

Calculation of (2.8) includes  $R_1 R_2$  scalar products of vectors in  $\mathbb{R}^n$ , leading to the overall complexity

$$\mathcal{N}_{\langle \cdot, \cdot \rangle} = O(dnR_1R_2).$$

3. The *Hadamard product*  $A \odot B \in \mathbb{R}^{\mathcal{I}}$  of two tensors  $A, B \in \mathbb{R}^{\mathcal{I}}$ ,  $A = [a_{\mathbf{i}}]$ ,  $B = [b_{\mathbf{i}}]$ , of the same size  $\mathcal{I}$  is defined componentwise

$$(A \odot B)_{\mathbf{i}} = a_{\mathbf{i}} b_{\mathbf{i}}, \quad \mathbf{i} \in \mathcal{I}.$$

Hence, for  $A_1, A_2$  given by (2.6) we tensorize the Hadamard product by

$$A_1 \odot A_2 := \sum_{k=1}^{R_1} \sum_{m=1}^{R_2} c_k b_m \left( u_k^{(1)} \odot v_m^{(1)} \right) \otimes \dots \otimes \left( u_k^{(d)} \odot v_m^{(d)} \right). \quad (2.9)$$

This leads to the complexity  $O(dnR_1R_2)$ .

4. In electronic structure calculations, the three-dimensional convolution transform with the Newton convolving kernel,  $p(x - y) = \frac{1}{\|x - y\|}$ , is the most computationally expensive operation. We employ the discrete version of the multi-dimensional convolution transform [13]

$$w(x) = \int_{\mathbb{R}^3} f(y)p(x - y)dy, \quad x \in \mathbb{R}^3, \text{ supp } f \subset [-b, b]^3,$$

by applying the standard collocation scheme to discretise the convolution product on a tensor grid

$$\omega_{\mathbf{3},n} := \omega_1 \times \omega_2 \times \omega_3, \quad \omega_\ell := \{-b + (m - 1)h : m = 1, \dots, n + 1\}, \quad \ell = 1, \dots, 3, \quad (2.10)$$

with mesh-size  $h = 2b/n$  with  $n$  being an even number. We denote the grid points by  $\{x_{\mathbf{m}}\}$ ,  $\mathbf{m} \in \mathcal{M} := \{1, \dots, n + 1\}^3$ . For given piecewise constant basis functions  $\{\phi_{\mathbf{i}}\}$ ,  $\mathbf{i} \in \mathcal{I} := \{1, \dots, n\}^3$ , associated with  $\omega_{\mathbf{3},n}$ , and a given continuous density function  $f$ , let  $f_{\mathbf{i}} = f(y_{\mathbf{i}})$  be the representation coefficients of  $f$  in  $\{\phi_{\mathbf{i}}\}$ ,

$$f(y) \approx \sum_{\mathbf{i} \in \mathcal{I}} f_{\mathbf{i}} \phi_{\mathbf{i}}(y), \quad (2.11)$$

where  $y_{\mathbf{i}}$  is the midpoint of the grid-cell (voxel)  $\delta_{\mathbf{i}} := \delta_{i_1} \times \delta_{i_2} \times \delta_{i_3}$  numbered by  $\mathbf{i} \in \mathcal{I}$ , with  $\delta_{i_\ell} := [-b + (i_\ell - 1)h, -b + i_\ell h]$  ( $\ell = 1, \dots, 3$ ). Now the collocation scheme reads as

$$f * p \approx \{W_{\mathbf{m}}\}_{\mathbf{m} \in \mathcal{M}}, \quad W_{\mathbf{m}} := \sum_{\mathbf{i} \in \mathcal{I}} f_{\mathbf{i}} \int_{\mathbb{R}^3} \phi_{\mathbf{i}}(y)p(x_{\mathbf{m}} - y)dy, \quad x_{\mathbf{m}} \in \omega_{\mathbf{3},n}.$$

As a first step, we precompute the coefficients

$$p_{\mathbf{i}} = \int_{\mathbb{R}^3} p(y)\phi_{\mathbf{i}}(y)dy, \quad \mathbf{i} \in \mathcal{I}.$$

The coefficient tensor  $P = [p_{\mathbf{i}}] \in \mathbb{R}^{\mathcal{I}}$  for the Coulomb potential  $p(x - y) = \frac{1}{\|x - y\|}$  is approximated in the rank- $R_N$  canonical tensor format using the optimised *sinc*-quadratures

[2], where the rank parameter  $R_N = O(|\log \varepsilon| \log n)$  depends logarithmically on both the required accuracy  $\varepsilon > 0$  and the grid size  $n$ . The 3rd order coefficient tensor  $F = [f_{\mathbf{i}}] \in \mathbb{R}^{\mathcal{I}}$  is approximated either in the rank  $\mathbf{r} = (r, r, r)$  Tucker format or via the canonical model with tensor rank  $R$ .

Following [13, 14], the resultant discrete convolution tensor  $[W_{\mathbf{m}}]$  can be obtained by copying the corresponding portion of the *tensor convolution* in  $\mathbb{V}_{\mathbf{n}}$ ,

$$F * P := [z_{\mathbf{j}}], \quad z_{\mathbf{j}} := \sum_{\mathbf{i} \in \mathcal{I}} f_{\mathbf{i}} p_{\mathbf{j}-\mathbf{i}+\mathbf{1}}, \quad \mathbf{j} \in \mathcal{J} := \{1, \dots, 2n-1\}^3, \quad (2.12)$$

centred at  $\mathbf{j} = \mathbf{n}$ , where the sum is over all  $\mathbf{i}, \mathbf{j} \in \mathcal{I}$ , which lead to legal subscripts for  $v_{\mathbf{j}-\mathbf{i}+\mathbf{1}}$ , i.e.,  $\mathbf{j} - \mathbf{i} + \mathbf{1} \in \mathcal{I}$ .

Approximating  $F$  in the rank- $R$  canonical format, see (2.2), enables us to compute  $F * P$  in the form (for two canonical tensors as in (2.6))

$$F * P := A_1 * A_2 = \sum_{k=1}^{R_N} \sum_{m=1}^R c_k b_m \left( u_k^{(1)} * v_m^{(1)} \right) \otimes \left( u_k^{(2)} * v_m^{(2)} \right) \otimes \left( u_k^{(3)} * v_m^{(3)} \right). \quad (2.13)$$

Assuming that one-dimensional convolutions  $u_k^{(\ell)} * v_m^{(\ell)} \in \mathbb{R}^{2n-1}$  can be computed in  $O(n \log n)$  operations, the complexity estimate takes the form

$$\mathcal{N}_{*} = O(n \log n R_N R).$$

As we already mentioned the tensor product convolution considerably outperforms the conventional 3D FFT having the complexity  $O(n^3 \log n)$ , see numerics in [15].

## 3 Calculation of the Hartree-Fock exchange

### 3.1 Agglomerated representation of the exchange operator

The exchange Galerkin matrix  $K_{ex}$  with respect to the normalized basis set  $\{g_k\}_{k=1, \dots, R_0}$  is given by

$$K_{ex} = \{K_{ij}\}_{i,j=1}^{R_0}, \quad K_{ij} := -\frac{1}{2} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} g_i(x) \frac{\tau(x, y)}{\|x - y\|} g_j(y) dx dy, \quad i, j = 1, \dots, R_0, \quad (3.1)$$

where the density matrix  $\tau(x, y)$  is defined as

$$\tau(x, y) = \sum_{a=1}^{N_{orb}} \varphi_a(x) \varphi_a(y),$$

over all occupied orbitals  $a$ .

The low cost of the three-dimensional convolution using the canonical representation of the convolving tensors makes possible the agglomerated numerical evaluation of the exchange matrix in the Fock operator. For this purpose, we divide the integration in (3.1) into the

following steps. First, we compute the convolutions of the pointwise products of molecular orbitals with the vectors from the normalized Gaussian basis set

$$W_{aj}(x) = \int_{\mathbb{R}^3} \frac{\varphi_a(y)g_j(y)}{\|x-y\|} dy \quad a = 1, \dots, N_{orb}, \quad j = 1, \dots, R_0. \quad (3.2)$$

These are further used for the calculation of contributions to the Galerkin matrix elements from every orbital  $a$ ,

$$V_{ij,a} = \int_{\mathbb{R}^3} \varphi_a(x)g_i(x)W_{aj}(x)dx, \quad i, j = 1, \dots, R_0. \quad (3.3)$$

The entries of the exchange matrix are then the sums of the corresponding values over all orbitals

$$K_{ij} = \sum_{a=1}^{N/2} V_{ij,a}, \quad i, j = 1, \dots, N/2. \quad (3.4)$$

We compute the exchange matrix (3.1) using the discrete tensor product representation of arising functions and operators.

The orbital of the molecule is considered as an expansion over the basis set of well separable continuous functions  $g_k(x)$ ,

$$\varphi_a(x) = \sum_{k=1}^{R_0} c_{a,k}g_k(x), \quad x = (x_1, x_2, x_3) \in \mathbb{R}^3, \quad (3.5)$$

where the basis functions  $g_k$ ,  $k = 1, \dots, R_0$ , are represented as the rank- $R$  canonical tensor products,

$$g_k(x) = \sum_{\nu=1}^R g_{k,\nu}^{(1)}(x_1) g_{k,\nu}^{(2)}(x_2) g_{k,\nu}^{(3)}(x_3), \quad (3.6)$$

with 1, 2, 3 designating spatial dimensions.

## 3.2 Discrete computational scheme

GTOs are used as conventional basis sets in electronic structure calculations due to their separability in spatial variables which is used in the analytical evaluation of the integrals in the calculation of the Hartree and exchange potentials.

In the following, for numerical illustrations, we choose the discretized Gaussians as vectors in the rank-1 canonical representations of the basis functions, mainly for the sake of convenient verification of the results of computations (the corresponding Galerkin matrix) with the standart MOLPRO output [26].

The rank-1 GTO basis functions  $g_k(x)$ ,  $k = 1, \dots, R_0$ , are given by (3.6) with  $R = 1$ , where  $g_{k,1}^{(\ell)}(x_\ell)$  denotes the generalized univariate Gaussians. The univariate Gaussians  $g_k^{(\ell)}(x_\ell) = g_{k,1}^{(\ell)}(x_\ell)$ ,  $\ell = 1, 2, 3$ , are the functions with infinite support given as

$$g_k^{(\ell)}(x_\ell) = (x_\ell - A_{\ell,k})^{p_{\ell,k}} \exp(-\alpha_k(x_\ell - A_{\ell,k})^2), \quad x_\ell \in \mathbb{R}, \quad \alpha_k > 0,$$

where  $p_{\ell,k} = 0, 1, \dots$  is the polynomial degree, and the points  $(A_{1,k}, A_{2,k}, A_{3,k}) \in \mathbb{R}^3$  specify the positions of nuclei in a molecule. In our scheme we use the *discrete* basis functions (given by vectors of the canonical tensor representation (2.2)) which are constructed by discretizing the Gaussians on the given tensor grid by using the associated piecewise constant basis functions.

Assume that the molecule is embedded in a certain fixed computational box  $[-b, b]^3$  with a suitable  $b > 0$ . For simplicity of notation, we take  $n_\ell = n$  equal for all dimensions. We introduce the equidistant tensor grid  $\omega_{\mathbf{3},n}$ , see (2.10) in §2. The grid points are denoted by  $\{x_{\mathbf{m}}\}$ ,  $\mathbf{m} \in \mathcal{M} := \{1, \dots, n+1\}^3$ . We use representation like (2.11) with  $f(x) = g_k(x)$ , where the rank-1 coefficients tensor  $G_k$  is given by the values of  $\ell$ -mode functions  $g_k^{(\ell)}$  at the centers  $y_{i_\ell}^{(\ell)}$  of intervals of the univariate grid  $[x_{i_\ell}^{(\ell)}, x_{i_\ell+1}^{(\ell)}]$ ,  $i_\ell = 1, \dots, n$ . This results in canonical vectors of length  $n$  with the entries  $\{g_k^{(\ell)}(y_{i_\ell}^{(\ell)})\}_{i_\ell=1}^n$ ,

$$\gamma_k^{(\ell)} = \{g_k^{(\ell)}(y_{i_\ell}^{(\ell)})\}_{i_\ell=1}^n \in \mathbb{R}^n, \quad \text{for } \ell = 1, 2, 3, k = 1, \dots, R_0, \quad (3.7)$$

such that  $G_k = \gamma_k^{(1)} \otimes \gamma_k^{(2)} \otimes \gamma_k^{(3)}$ . By summing tensor products of the canonical vectors with the corresponding weights  $c_{a,k}$  as in (2.2) we obtain the discrete representation of the orbital  $\varphi_a$ ,  $a = 1, \dots, N_{orb}$ , in the rank- $R_0$  canonical format,

$$U_a = \sum_{k=1}^{R_0} c_{a,k} \gamma_k^{(1)} \otimes \gamma_k^{(2)} \otimes \gamma_k^{(3)}, \quad c_{a,k} \in \mathbb{R}, \quad (3.8)$$

where  $R_0$  is the number of basis functions. This discretization can be considered as a representation in the Galerkin set of basis functions  $\{G_k\}$  obtained by representing the initial continuous basis set  $\{g_k\}$  via piecewise constant basis functions  $\{\phi_i\}$  on the uniform grid (see (3.7)).

We use the rank- $R_N$  canonical tensor product representation of the coefficient tensor  $P$  for the Newton potential  $\frac{1}{\|x-y\|}$ , on the same grid. This tensor is precomputed by using the optimized sinc-quadratures [2, 13], where the rank parameter  $R_N = O(|\log \varepsilon| \log n)$  depends logarithmically on both the required accuracy  $\varepsilon > 0$  and the univariate grid size  $n$ . In particular, for our computations the tensor  $P$ , representing the Newton potential has the canonical rank in the range  $20 \leq R_N \leq 30$ , depending on the one-dimension grid size  $n$  and accuracy requirements  $\varepsilon > 0$ .

We present Algorithm 1 describing the computational scheme for evaluation of (3.2) - (3.4) in tensor product format<sup>2</sup>.

**Lemma 3.1** *The complexity of Algorithm 1 for the computation of the exchange Galerkin matrix  $K_{ex}$  in the Hartree-Fock equation using the discretized GTO basis is estimated by*

$$W_{K_{ex}} = O(N_{orb} R_N (R_0^2 n \log n + R_0^4 n_{ef})).$$

---

<sup>2</sup>The Hadamard product  $\theta_{a,j} = U_a \odot G_j$  in the Algorithm 1 can be either (1) stored for all vectors  $\gamma_k$  at step **(A)** or (2) recomputed before evaluation of the scalar products at step **(C)**. Due to very low cost of this operation, and large storage requirements for the case of large grids,  $O(R_0^2 n)$ , we prefer the case (2).

---

**Algorithm 1** Computation of the Exchange Matrix in Tensor Arithmetics

---

*Input data:* rank- $R_0$  canonical tensors  $U_a \in \mathbb{V}_n$ ,  $a = 1, \dots, N_{orb}$ , rank  $R_N$  tensor  $P \in \mathbb{V}_n$ , rank-1 canonical tensors  $G_k = \gamma_k^{(1)} \otimes \gamma_k^{(2)} \otimes \gamma_k^{(3)}$ ,  $k = 1, \dots, R_0$ , and the filtering threshold  $\varepsilon_F > 0$ .

(A0) Find effective supports  $\sigma_j \subset [-b, b]$  for  $\gamma_j$ ,  $j = 1, \dots, R_0$ , by  $\varepsilon_F$ -thresholding,

$$\sigma_j = \sigma_j^{(1)} \times \sigma_j^{(2)} \times \sigma_j^{(3)}, \text{ where } \sigma_j^{(\ell)} = \{i : |\gamma_j^{(\ell)}(x_i)| \geq \varepsilon_F\} \subset \{1, \dots, n\}, \quad \ell = 1, 2, 3.$$

**for**  $a = 1, \dots, N_{orb}$

**for**  $k = 1, \dots, R_0$

(A) Compute the Hadamard product  $\theta_{a,k} = U_a \odot G_k$  of tensors  $U_a$  and  $G_k$  by using (2.9).

(B) Compute the tensor convolution  $\Theta_{a,k} = \theta_{a,k} * P$  by using (2.13).

**for**  $j = 1, \dots, R_0$

(C) Compute the restricted scalar products in the window  $\sigma_j$ ,

$$K_{a,k,j} = \langle \theta_{a,j}, \Theta_{a,k} \rangle_{|\sigma_j},$$

**end for**  $j$

**end for**  $k$

**end for**  $a$ .

(D) Sum matrix elements over all orbital indices,  $K_{kj} = \sum_{a=1}^{N_{orb}} K_{a,k,j}$ , for  $k, j = 1, \dots, R_0$ .

*Output data:* the exchange matrix  $K = \{K_{kj}\}_{k,j=1}^{R_0}$ .

---

*Proof.* This estimate includes the cost of the evaluation of convolutions in (3.2) for every orbital,  $O(N_{orb}R_N R_0^2 n \log n)$ , and the scalar product (3.3) of the tensor  $\Theta_{a,k}$  with the products of the orbitals and Gaussians,  $O(N_{orb}R_N R_0^4 n_{ef})$ . ■

Since the canonical rank  $R_N$  of tensor  $P$  corresponding to the Coulomb potential depends only logarithmically on  $n$ , it can be treated as a constant.

**Remark 3.2** Notice that the rank reduction of the canonical tensor  $\Theta_{a,k}$  after step (3.2) reduces the complexity to

$$W_{K_{ex,red}} = O(N_{orb}R_0^3 n_{ef}). \quad (3.9)$$

In the case of large molecules further optimization up to  $O(N_{orb}R_0^2 n_{ef})$ -complexity is possible due to the rank reduction applied to the rank- $R_0$  orbitals (tensors  $U_a$ ).

**Remark 3.3** The rank- $R_0$  tensors  $U_a$ ,  $a = 1, \dots, N_{orb}$ , representing the orbitals, can be chosen as the Galerkin basis set  $\{G_a\}$ ,  $a = 1, \dots, N_{orb}$ , where  $N_{orb}$  is usually much smaller than  $R_0$ . This may relax the critical dependence  $O(R_0^4)$  as in Lemma 3.1 above (see also Lemma 3.1 in [16]).

### 3.3 Rank reduction

The maximal initial rank of tensor  $\Theta_{a,k}$  at the step (B) in Algorithm 1 is given by  $R_\Theta = R_N R_0$ . We perform the rank reduction for this tensor by the canonical-to-Tucker (C2T)

|  | CH <sub>4</sub> | CH <sub>3</sub> OH | C <sub>2</sub> H <sub>5</sub> OH |
|--|-----------------|--------------------|----------------------------------|
| $R_{\Theta}$                                 | 1250            | 1875               | 2775                             |
| $r_T = 12, \epsilon_T \leq 10^{-7}, R_{RED}$ | 80              | 90                 | 110                              |
| $coef_R$                                     | 15              | 20                 | 23                               |
| $r_T = 10, \epsilon_T \leq 10^{-6}, R_{RED}$ | 50              | 70                 | 100                              |
| $coef_R$                                     | 25              | 26                 | 27                               |

Table 3.1: Rank reduction for  $\Theta_{a,k}$  in computation of the exchange matrix for the pseudopotential case of some molecules.

and Tucker-to-canonical (T2C) algorithm introduced and discussed in details in [9, 15]. In particular, it is shown that the multigrid version of the C2T algorithm applied to the 3-rd order rank- $R$  canonical tensors has linear complexity with respect to all parameters of the input tensor: the canonical rank  $R$ , the Tucker rank  $r$ , and the univariate grid size  $n$ . Thus, we can reduce the complexity of Algorithm 1 to (3.9) by solely multilinear algebraic methods, which do not take into account any previous knowledge on the molecular structure.

Table 3.3 shows the average rank reduction by the C2T and T2C algorithms applied to tensor  $\Theta_{a,k}$  in calculations for the molecules CH<sub>4</sub>, CH<sub>3</sub>OH and C<sub>2</sub>H<sub>5</sub>OH. We present the approximate canonical ranks  $R_{RED}$  (and respective Tucker ranks  $r_T$ ) of the tensors corresponding to the *largest value*, over the parameters  $a = 1, \dots, N_{orb}$ ,  $k = 1, \dots, R_0$ , to achieve the prescribed approximation error  $\epsilon_T$ ,

$$R_{RED} = \max_{1 \leq a \leq N_{orb}, 1 \leq k \leq R_0} R_{RED}(a, k),$$

where  $R_{RED}(a, k)$  denotes the reduced canonical rank of  $\Theta_{a,k}$ , for a given  $\varphi_a$  and  $g_k$ . Table 3.3 gives also the corresponding reduction coefficient,  $coef_R = \frac{R_{\Theta}}{R_{RED}}$ .

### 3.4 Window technique for fast computation of the inner products

We compute the algebraic tensor representation of the discrete electron orbitals  $U_a$  given by (3.8) using the coefficients of their representation in the discrete Gaussian basis set  $\gamma_k^{(\ell)}$ . It turns out by the construction that most of  $\gamma_k^{(\ell)}$  have local character (fast exponential decay) with respect to the size of the whole computation domain  $[-b, b]^3$ . Therefore we precompute effective supports of the canonical vectors  $\gamma_k^{(\ell)}$  by truncating their parts, which are lower than some predefined threshold  $\varepsilon > 0$ . We call it the “windowing” procedure for finding the active interval for each Gaussian. In our case, the resulting effective vector size of the canonical vectors is at average 3 times smaller than the corresponding grid size  $n$  even for small molecules. The resulting “effective” univariate grid size is  $n_{ef} = \alpha n$ , with  $\alpha = \alpha(\varepsilon) < 1$ . For example, for small molecules  $\alpha \sim 0.2 \div 0.3$  for  $\varepsilon = 10^{-5}$ . This leads to reduced cost of the scalar products with respect to the univariate grid size  $n$ .

We expect much stronger windowing effect in the case of large molecules, while it can be directly applied to the Hadamard products  $U_a \odot G_k$ .

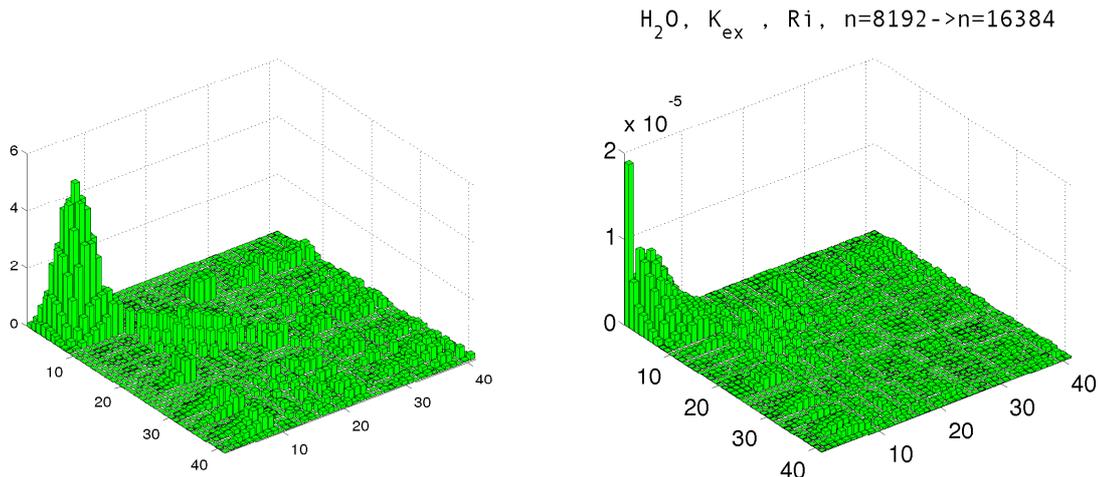


Figure 4.1: Left: the entries of exchange matrix for all electron case of H<sub>2</sub>O. Right: absolute error of the tensor-structured computation on the grids with  $n = 8192$  and  $n = 16384$ .

## 4 Numerical experiments

We tested the presented tensor-structured method, by the computation of the exchange Galerkin matrix for the following molecules:

- all electron case : H<sub>2</sub>O ( $N_{orb} = 5$ ,  $R_0 = 41$ ), CH<sub>4</sub> ( $N_{orb} = 5$ ,  $R_0 = 55$ );
- pseudopotential case: CH<sub>4</sub> ( $N_{orb} = 4$ ,  $R_0 = 50$ ), CH<sub>3</sub>OH ( $N_{orb} = 7$ ,  $R_0 = 75$ ) and C<sub>2</sub>H<sub>5</sub>OH ( $N_{orb} = 11$ ,  $R_0 = 111$ ).

The calculations are performed on a standard SUN station using Matlab 7.6. In presented figures we give the absolute difference of the results of our computations with the corresponding exchange matrix calculated by the MOLPRO program [26].

The computational box  $[-b, b]^3$  for small molecules is in the range of  $2b = 14 \text{ \AA}$  for H<sub>2</sub>O, and  $2b = 20 \text{ \AA}$  for CH<sub>4</sub>, C<sub>2</sub>H<sub>5</sub>OH, and CH<sub>3</sub>OH. The mesh-size varies from  $h \approx 2 \cdot 10^{-2}$  to  $8 \cdot 10^{-4} \text{ \AA}$ .

### 4.1 All electron case

For the molecules with moderate size  $R_0$  of basis sets, like CH<sub>4</sub> or H<sub>2</sub>O the grid-sizes up to  $n \sim 1.6 \cdot 10^4$  are computationally feasible for MATLAB, which is equivalent to computations with  $n^3 \sim 10^{12}$  nodes in the volume. These grids enable accurate computations for all electron case, resolving strong cusps corresponding to the core electrons, see Figures 4.2 and 4.1.

Computation of the exchange Galerkin matrix for all electron case of H<sub>2</sub>O molecule is a challenging problem due to “sharp” Gaussians corresponding to core electrons of the Oxygen atom which should be resolved using fine grids. Our technique solves this problem by enabling fine grids up to  $n = 16384$ . Figure 4.1 (left) shows the absolute values of

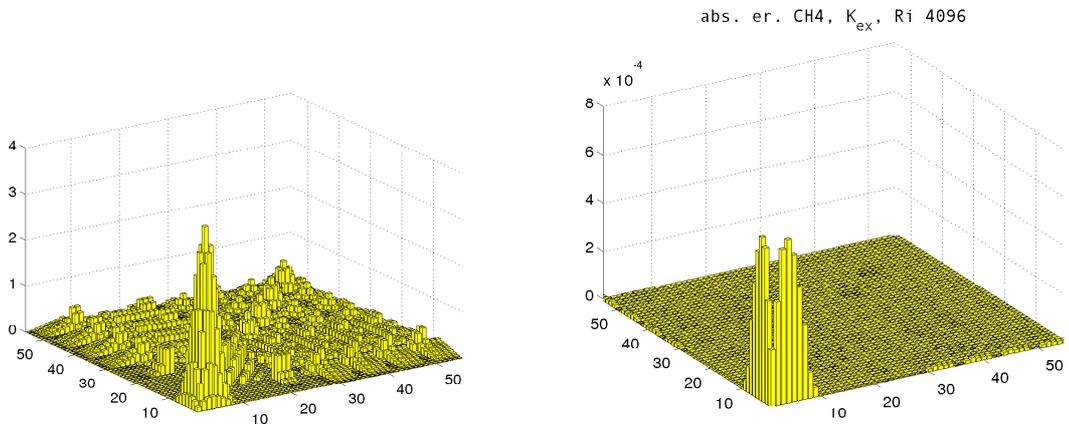


Figure 4.2: Left: the entries of exchange matrix for  $\text{CH}_4$ . Right: absolute approximation error of the tensor-structured computations on the grids with  $n = 2048$  and  $n = 4096$ .

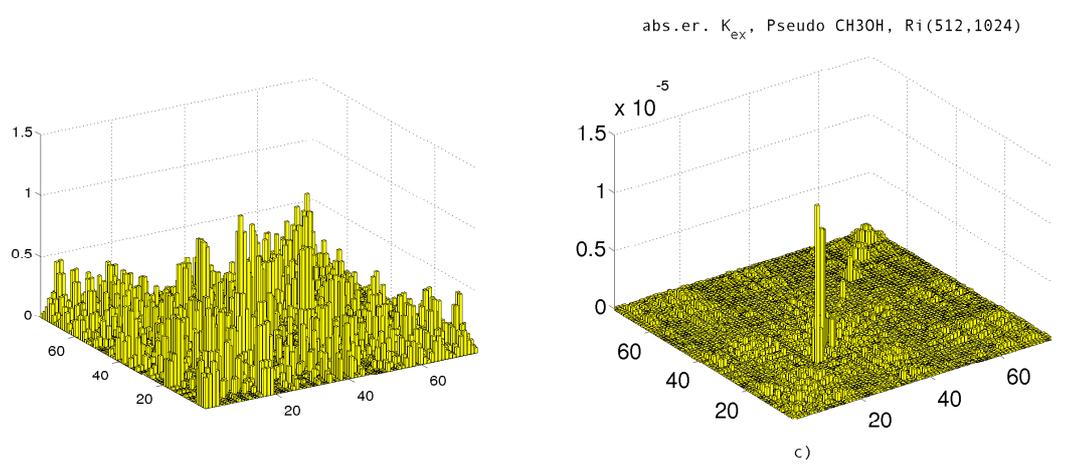


Figure 4.3: Left: the exchange matrix for the *pseudopotential* case of  $\text{CH}_3\text{OH}$ . Right: absolute approximation error of the tensor-product computation on the grids with  $n = 512$  and  $n = 1024$ .

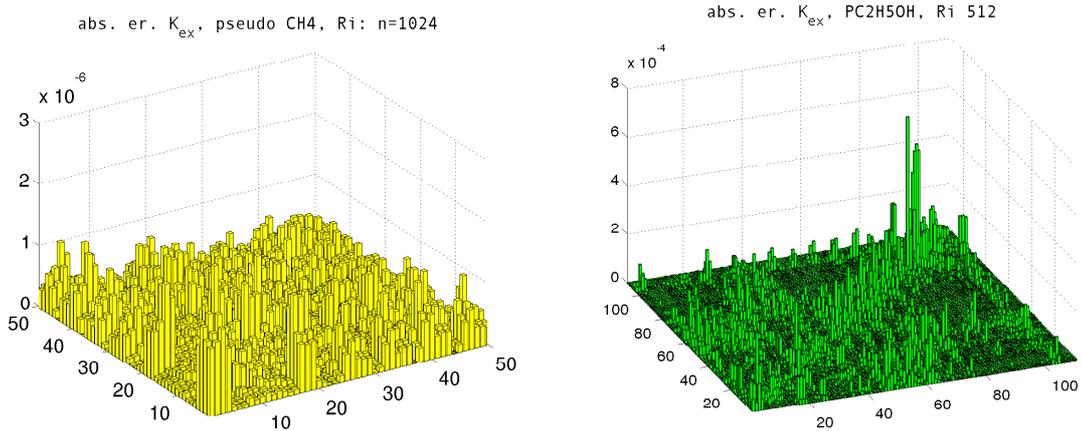


Figure 4.4: a) Absolute error in the tensor-product computation of the exchange matrix for the pseudopotential case of molecules  $\text{CH}_4$  (left) and  $\text{C}_2\text{H}_5\text{OH}$  (right).

| $n^3$                       | $64^3$ | $128^3$ | $256^3$ | $512^3$ | $1024^3$ |
|-----------------------------|--------|---------|---------|---------|----------|
| $\text{H}_2\text{O}$        | 1      | 1.3     | 2.0     | 3.2     | 8.0      |
| $\text{CH}_4$ (ps)          | 1      | 1.3     | 2.0     | 3.6     | 8.9      |
| $\text{CH}_3\text{OH}$ (ps) | 1      | 1.3     | 1.9     | 3.3     | 5.1      |

Table 4.1: Comparison of relative times.

the exchange matrix entries, Figure 4.1 (right) shows that the absolute error of the tensor-structured computations using the Richardson extrapolation on the grids with  $n = 8192$  and  $n = 16384$ . We reach the accuracy  $1.89 \cdot 10^{-5}$  in the “cusp area”, the remaining entries are computed with the absolute error in the range of  $10^{-6} \div 10^{-8}$ .

Figure 4.2 (left) shows the absolute values of the exchange matrix of  $\text{CH}_4$  and Figure 4.2 (right) shows the absolute error of tensor-structured computations reaching the accuracy  $10^{-4}$  by using the Richardson extrapolation on the grids with  $n = 2048$  and  $n = 4096$ . Again, the matrix entries apart from the “cusp area” are computed with much higher accuracy.

## 4.2 Pseudopotential case

We consider the pseudopotential case for larger molecules, achieving the accuracy up to  $10^{-6}$ , using smaller grid sizes up to  $n = 1024$ . The Fortran version of the loops including steps (C) – (D) in Algorithm 1 can improve dramatically the CPU computation time.

Figure 4.3 (left) shows entries of the exchange matrix of  $\text{CH}_3\text{OH}$  molecule and Figure 4.3 (right) shows that the absolute error of tensor-structured computations for this molecule using the Richardson extrapolation on grids with  $n = 512, 1024$  yield the accuracy  $\sim 10^{-5}$ . Figure 4.4 shows the absolute error of the tensor-structured computations for the exchange matrices in the pseudopotential case of the  $\text{CH}_3\text{OH}$  molecule (left) and  $\text{C}_2\text{H}_5\text{OH}$  (right), correspondingly. For  $\text{CH}_3\text{OH}$  the Richardson extrapolation on two consequent grids with  $n = 512, 1024$  yields the accuracy  $\sim 10^{-5}$ , while for  $\text{C}_2\text{H}_5\text{OH}$  we obtain  $7 \cdot 10^{-4}$  already on

the grids  $n = 256, 512$ .

Table 4.2 shows linear scaling of relative computation times with respect to the one-dimension grid size  $n$ , in respective units of the coarsest grid calculations ( $n = 64$ ) for one orbital.

These calculations show that the tensor-structured methods on 3D Cartesian grids give promising results for their application in computation of the multivariate integrals in quantum chemistry. These can include perspectives of computations either for large molecules applying discretization on fine Cartesian grids, or in the framework of the post-Hartree-Fock models.

**Acknowledgements.** The author cordially acknowledges Dr. Heinz-Jürgen Flad (TU Berlin) and PD DrSci. Boris Khoromski (MIS MPI Leipzig) for the formulation of the problem and valuable discussions at all stages of this work. The author gratefully acknowledges Prof. Dr. R. Schneider (TU Berlin) for useful discussions.

## References

- [1] E. Acar, T. G. Kolda and D. M. Dunlavy. An Optimization Approach for Fitting Canonical Tensor Decompositions. Technical Report Number SAND2009-0857, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, February 2009.
- [2] C. Bertoglio, B. N. Khoromskij. Low rank tensor-product approximation of projected Green kernels via sinc-quadratures. Preprint 79/2008, MPI MIS Leipzig, 2008 (submitted).
- [3] L. De Lathauwer, B. De Moor, J. Vandewalle. *On the best rank-1 and rank- $(R_1, \dots, R_N)$  approximation of higher-order tensors*. SIAM J. Matrix Anal. Appl., 21 (2000) 1324-1342.
- [4] L. De Lathauwer, B. De Moor, J. Vandewalle. *A multilinear singular value decomposition*. SIAM J. Matrix Anal. Appl., 21 (2000) 1253-1278.
- [5] I.P. Gavriljuk, W. Hackbusch and B.N. Khoromskij. Hierarchical Tensor-Product Approximation to the Inverse and Related Operators in High-Dimensional Elliptic Problems. Computing 74 (2005), 131-157.
- [6] I. P. Gavriljuk, W. Hackbusch and B. N. Khoromskij. Data-sparse approximation of a class of operator-valued functions. Math. Comp. 74 (2005), pp. 681-708.
- [7] H.-J. Flad, W. Hackbusch, B.N. Khoromskij, and R. Schneider. *Concept of Data-Sparse Tensor-Product Approximation in Many-Particle Modeling*. In: “Matrix Methods: Theory, Algorithms, Applications”, pp. 313-347, World Scientific publishing, Singapore, 2010.
- [8] W. Hackbusch, B.N. Khoromskij and E.E. Tyrtshnikov. *Hierarchical Kronecker tensor-product approximations*. J. Numer. Math. **13** (2005), 119–156.

- [9] V. Khoromskaia. Numerical Solution of the Hartree-Fock equation by the Tensor-Structured Methods. PhD Dissertation, in preparation, MPI MiS, Leipzig, 2010.
- [10] B. N. Khoromskij. *Structured Rank- $(r_1, \dots, r_d)$  Decomposition of Function-related Tensors in  $\mathbb{R}^d$* . Comp. Meth. in Applied Math., **6** (2006), 2, 194-220.
- [11] B. N. Khoromskij. On Tensor Approximation of Green Iterations for Kohn-Sham equations. Computing and Visualisation in Science, (2008) 11:259-271.
- [12] B. N. Khoromskij. Tensor-structured Preconditioners and Approximate Inverse of Elliptic Operators in  $\mathbb{R}^d$ . J. Constructive Approximation, **30**:599-620 (2009).
- [13] B. N. Khoromskij. *Fast and Accurate Tensor Approximation of Multivariate Convolution with Linear Scaling in Dimension*. Preprint MPI MIS 36, 2008, Leipzig; J. Comp. Appl. Math., 2010, DOI: 10.1016/j.cam.2010.02.004, to appear.
- [14] B. N. Khoromskij and V. Khoromskaia. *Low Rank Tucker Tensor Approximation to the Classical Potentials*. Central European J. of Math., **5**(3) 2007, 1-28.
- [15] B.N. Khoromskij and V. Khoromskaia. *Multigrid Tensor Approximation of Function Related Arrays*. SIAM J. on Sci. Comp., **31**(4), 3002-3026 (2009).
- [16] B.N. Khoromskij, V. Khoromskaia and H.-J. Flad. *Numerical solution of the Hartree-Fock equation in multilevel tensor-structured format*. Preprint 44/2009 MPI MIS, Leipzig, 2009.
- [17] B. N. Khoromskij, V. Khoromskaia, S. R. Chinnamsetty, H.-J. Flad. *Tensor Decomposition in Electronic Structure Calculations on 3D Cartesian Grids*. J. of Comput. Phys. **228** (2009) 5749-5762.
- [18] T. G. Kolda and B. W. Bader, *Tensor Decompositions and Applications*. SIAM Rev., **51**(2009), no. 3, pp. 455-500.
- [19] C. Le Bris, ed.. *Handbook of Numerical Analysis, Vol. X, Computational Chemistry*, North-Holland, 2003.
- [20] C. Le Bris. *Computational chemistry from the perspective of numerical analysis*. Acta Numerica (2005), 363 - 444.
- [21] I. Oseledets, D. Savostyanov, E. E. Tyrtysnikov. *Cross approximation in electron density computations*. Num. Lin. Alg. Appl., submitted, 2009.
- [22] I. Oseledets, D. Savostyanov, E. E. Tyrtysnikov. *Linear algebra for tensor problems*, Computing, **85** (2009), 169-188.
- [23] R. Polly, H.-J. Werner, F. R. Manby and P. J. Knowles. Fast Hartree-Fock theory using density fitting approximations. Mol.Phys. **102**:2311-2321, 2004.
- [24] J. Sielk, H. F. von Horsten, Fritz Krüger, Reinhold Schneider, B. Hartke. Quantum-mechanical wavepacket propagation in a sparse, adaptive basis of interpolating Gaussians with collocation. Physical Chemistry Chemical Physics, 2009, **11**, pp. 463-475.

- [25] J. VandeVondele, M. Krack, F. Mohamed, M. Parrinello, Th. Chassaing, J. Hutter. QUICKSTEP: Fast and accurate density functional calculations using a mixed gaussian and plane waves approach. *Comp. Phys. Comm.*, 167(2005) 103-128.
- [26] H.-J. Werner, P.J. Knowles et al. MOLPRO, version 2002.10, a package of *ab initio* programs for electronic structure calculations.
- [27] T. Yanai, G. Fann, Z. Gan, R. Harrison and G. Beylkin. Multiresolution quantum chemistry: Hartree-Fock exchange. *J. Chem. Phys.* 121 (14) (2004) 6680-6688.