

# UTChem – A Program for *ab initio* Quantum Chemistry

Takeshi Yanai<sup>1</sup>, Haruyuki Nakano<sup>2</sup>, Takahito Nakajima<sup>2,3</sup>, Takao Tsuneda<sup>2</sup>, So Hirata<sup>4</sup>, Yukio Kawashima<sup>5</sup>, Yoshihide Nakao<sup>2</sup>, Muneaki Kamiya<sup>2</sup>, Hideo Sekino<sup>6</sup>, and Kimihiko Hirao<sup>2,5</sup>

<sup>1</sup> Oak Ridge National Laboratory, PO Box 2008 MS6367  
Oak Ridge TN 37831-6367, USA

<sup>2</sup> Department of Applied Chemistry, School of Engineering,  
University of Tokyo, Tokyo, Japan 113-8656

<sup>3</sup> PREST, Japan Science and Technology Corporation (JST)

<sup>4</sup> William R Wiley Environmental Molecular Sciences Laboratory  
Battelle, Pacific Northwest National Laboratory  
K1-96, P.O.Box 999, Richland, WA 99352, USA

<sup>5</sup> Institute for Molecular Science  
Myodaiji, Okazaki, Japan 444-8585

<sup>6</sup> Department of Knowledge-based Information Engineering  
Toyohashi University of Technology  
Toyohashi, Japan 441-8580

**Abstract.** UTChem is a quantum chemistry software developed by Hirao's group at the University of Tokyo. UTChem is a research product of our work to develop new and better theoretical methods in quantum chemistry.

## 1 Introduction

Present molecular quantum theory is highly sophisticated, and has evolved dramatically. Software forms a basis for computational chemistry. However, it is not an easy task for an individual/group to develop a comprehensive new program package in *ab initio* quantum chemistry from scratch. Several years ago, we decided to accept this challenge. In view of the availability of such good programs as Gaussian, Gamess, Molcas, NWChem, etc., one may question the relevance of a new program package. We have three arguments for our project. (1) First, we believe that healthy competition is very important in science. (2) Second, we can have a good harvest by doing research using other programs, but it is an abortive flower. We could not make a true breakthrough if we were circumscribed by current software limitations. (3) Third, in spite of the excellent performance of other programs, there are important and powerful methods that others cannot yet handle. We have developed new methodologies in quantum chemistry, particularly the multireference-based perturbation theory for describing chemical reactions and excited states, relativistic molecular theory to treat heavy elements, parameter-free (less) exchange and correlation functionals in DFT, highly efficient algorithms for calculating molecular integrals over generally contracted Gaussians, etc. UTChem is a research product of our work to develop new and better theoretical methods in quantum chemistry. Most of the codes have been developed recently by Hirao's group at the

University of Tokyo. The basic philosophy behind UTChem is to develop methods that allow an accurate and efficient computational chemistry of electronic structure problems for molecular systems in both the ground and excited states. We have started joint developments with a few coalition partners, who are contributing their codes to UTChem. Their contributions have enhanced the capabilities of UTChem to carry out well-developed methods such as RPA (by Sekino), MPn, CI, CC, etc (by Hirata on TCE project),[1] which are standard in most quantum chemistry programs. We are aiming ultimately at better performance than other programs. UTChem will soon be ready for distribution. Here you will be able to see the features of UTChem. UTChem contains a large number of improvements and some interesting new features, which others cannot match.

## 2 Synthetic Design and Conception for Implementation of UTChem

A packaging of UTChem got started in a situation that researchers and students in Hirao laboratory had accomplished their implementations individually on independent developmental environments. The accomplishments are covering wide range of capabilities enough to do a sequence of *ab initio* calculations without using other extant distributions as follows,

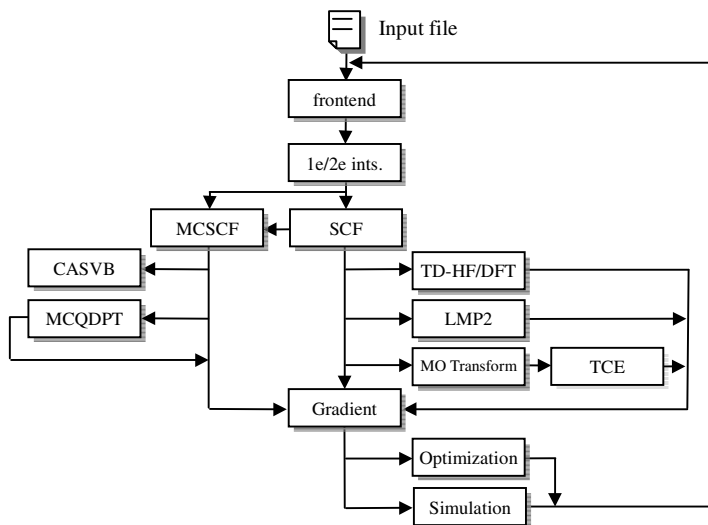
- 1) multireference perturbation method (MRMP, MCQDPT),
- 2) multiconfigurational self-consistent field method (QCAS, GMC, CASVB),
- 3) relativistic 1/2-component method (RESC, DK3),
- 4) relativistic 4-component method (DHF, DKS),
- 5) one- and two-electron integrals,
- 6) direct and conventional self-consistent field method,
- 7) density functional theory (OP, PFREE),
- 8) dynamics and simulation involving QM/MM,
- 9) response theory for Hartree-Fock and density functional theory.

Our synthetic design and conception are basically coming from a practical necessity on how smooth and stressless compilation of the above existing program codes can be carried out. The fundamental structure we desire is that UTChem should be a program suite which enables developer groups to develop separately executable programs of their owns individually with no consolidation into one binary. This approach is based on a development philosophy that programmer groups can program as freely as possible with no disturbance of serious problems and fatal bugs of the damaged codes the other programmer groups have committed, and no intruder of the other developer groups into their own pretty codes and motivations.

Technically, through a sequence of calculations using separated binaries, what the individual program binaries share with each other is data files compliant with the file formats the developers have specified. UTChem itself just governs a number of rules of the file interfaces, (i.e., no subroutine interfaces, no program rules, ...), which are an exclusive way to connect the individual executable binaries. This design is simple and widely used in other program systems, meanwhile a method to control several program binaries has to be implemented so as to distribute them as a reliable and usable program suite. In UTChem, Python language is used to provide a high-level

and flexible script to control and manage the calculation flows consisted of separately executed programs. We believe that this packaging approach also widens accessibility, in which various developers also easily contribute their own program to UTChem.

As to practical calculation flows, figure 1 depicts a flowchart in UTChem. After the frontend program reads a given input file, required one- and two-electron AO integrals are calculated. HF/KS-SCF as well as TD-HF/DFT for excitation energies and Local MP2 are carried out with 2e AO integrals directly-driven or restored from disk. The single-reference electron correlation theories such as configuration interaction theory (CISD, CISDT, CISDTQ), coupled-cluster theory (CCD, LCCD, CCSD, LCCSD, QCISD, CCSDT, CCSDTQ), and Møller-Plesset perturbation theory (MP2, MP3, MP4) achieved by tensor contraction engine (TCE) are following the integral transformation. Multiconfigurational calculations including MCSCF and perturbation (MRMP and MCQDPT) method are connected with AO integral part and HF-SCF part. After the above single-point energy calculations, if necessary and executable, the derivative energies with respect to the nuclear coordinates are derived. The gradients are used in the geometry optimization code and dynamics simulation including QM/MM.



**Fig. 1.** Schematic calculation flowchart in UTChem

## 2.1 INTEGRA

INTEGRA is currently a primal module of UTChem program suite, which we started to write from scratch in Jun 1999. Fortran 90 language is chosen for writing codes in point of its simplicity and flexibility even for quantum chemists reluctant to overcome the computational complexity. Our scientific motivation of the initial implementation of INTEGRA was to develop an efficient integral method for generally contracted Gaussian-type orbitals (GTOs). The successful achievement of the correlation consistent basis sets developed by Dunning[2] and the atomic natural orbital (ANO) basis sets by Almlöf and Taylor[3] has spurred demands of the efficient two-electron integral program based on the general contraction scheme. The difficulty in efficient calculations using general contraction scheme is that the generally contracted GTOs are in nature composed of quite a few primitive GTOs, and the conventional *ab initio* programs and algorithms are designed to calculate the segmentally contracted GTOs. Recently Ishida has reported an efficient formulation for electron repulsion integrals (ERIs) using accompanying coordinate expansion (ACE) method.[4] His formulation can yield a formulation of ERIs requiring the lowest FLOP counts in the ever reported formulations, and especially the reduction of FLOPs for ERIs over highly contracted GTOs is outstanding. We have developed a new integral method oriented to the generally contracted Gaussian-type orbitals (GTOs), and succeeded to implement an efficient integral code SPHERICA onto INTEGRA.[5] Our algorithm successfully specialized to the general contraction scheme with exploiting the efficiency of Ishida's formulation could lead to FLOP count dependence against the degree of contraction  $K$  and the number of contracted GTOs  $N$  as

$$\text{FLOPs} = x_1 K^4 N + x_2 K^3 N^2 + y_1 K^2 N^3 + y_2 K N^4 + z N^4. \quad (1)$$

Comparing with the segmental contraction scheme, which gives FLOP count dependence as,

$$\text{FLOPs} = N^4 (x K^4 + y K^2 + z), \quad (2)$$

our algorithm gains a drastic efficiency in  $O(N^3 + KN^2)$ . The name of the code SPHERICA is coming from a capability to calculate ERIs over spherical harmonics generally contracted GTOs efficiently without transformation from Cartesian indices. Table I shows the CPU times to calculate benzene molecule using ANO basis sets on the single CPU of IBM RS/6000 260 model. Currently the integral code SPHERICA is contributing to a direct and conventional SCF routine, in which a robust orbital optimization using direct inversion in the iterative subspace, DIIS, the symmetry usage within  $D_{2h}$  and subgroup, DFT numerical grid-quadrature routine for the exchange-correlation potentials, and the parallelization using Global Array[6] have been implemented.

Historically INTEGRA has got an integrated platform for other implementations toward post-SCF routine, such as the integral transformation routine, the time-dependent Hartree-Fock / density functional theory (TD-HF/DFT) routines, the single-reference many-electron correlation routines achieved by TCE, and so on, except MR2D and S&D. The relativistic calculation code REL4D is also programmatically included in INTEGRA. Even though several applications are

consolidated into one binary INTEGRA, a switch specified in a command option allows each application to execute independently, and therefore schematically each application can be regarded as an independent binary. The reason why developers have been working to write codes mainly in INTEGRA without getting their codes an individual binary is just for convenience of the program interface of INTEGRA and Fortran90 environment. Remarkably, the contributors of RPA and TCE were willing to work on cooperative coding on INTEGRA.

To summarize, INTEGRA efficiently provides the following basic integrals necessary for *ab initio* calculations with Cartesian and spherical harmonics basis,

- 1) one-electron integrals and first derivatives (overlap, core Hamiltonian and first dipole moment)
- 2) two-electron integrals and first derivatives,
- 3) one- and two-electron integrals transformed into MO indices.

The parallelized HF/KS-SCF and parallelized TD-HF/DFT can be calculated using directly driven integrals. The detail features on DFT, TCE and REL4D modules are mentioned in other sections.

**Table 1.** CPU times (sec.) to calculate Fock matrix in SCF calculation for benzene molecule.

	Primitive / Contracted GTOs	Number of basis	SPHERICA	HONDO	MOLCAS
ANO-S	C: 10s6p3d / 3s2p1d H: 7s3p / 2s1p	114	379	2,710	237
ANO-S	C: 10s6p3d / 4s3p2d H: 7s3p / 3s2p	192	870	17,500	358
ANO-L	C: 14s9p4d / 3s2p1d H: 8s4p / 2s1p	114	1,003	4,330	796
ANO-L	C: 14s9p4d3f / 4s3p2d1f H: 8s4p3d / 3s2p1d	264	6,080	125,000	4,910

## 2.2 MR2D – Multi-Reference 2nd-Order Perturbation Theory and Its Derivatives

MR2D is a program for a multistate multireference perturbation theory, the quasi-degenerate perturbation theory with MC-SCF reference functions (MC-QDPT) [7]. In this PT, state-averaged CAS-SCF is first performed to set reference functions, and then an effective Hamiltonian is constructed, which is finally diagonalized to obtain the energies of interest. This theory includes multireference Møller–Plesset (MRMP) PT (the case that the set of reference functions reduces to a single function).[8]

Let the reference functions be CAS-SCF wavefunctions,

$$|\alpha\rangle = \sum_A C_A |A\rangle. \quad (3)$$

The effective Hamiltonian to second order is given by

$$\left(H_{\text{eff}}^{(0-2)}\right)_{\alpha\beta} = \langle\alpha|H|\beta\rangle + \frac{1}{2} \sum_I \left\{ \frac{\langle\alpha|H|I\rangle\langle I|H|\beta\rangle}{E_\beta^{(0)} - E_I^{(0)}} + \frac{\langle\beta|H|I\rangle\langle I|H|\alpha\rangle}{E_\alpha^{(0)} - E_I^{(0)}} \right\}. \quad (4)$$

Substituting the second-quantized operator into  $H$ , we obtain an explicit formula using molecular integrals and orbital energies instead of matrix elements,

$$\begin{aligned} \left(H_{\text{eff}}^{(0-2)}\right)_{\alpha\beta} = & E_\alpha^{\text{CAS}} \delta_{\alpha\beta} - \sum_{pq,B} \langle\alpha|E_{pq}|B\rangle C_B(\beta) \sum_e \frac{u_{pe} u_{eq}}{\epsilon_e - \epsilon_q + \Delta E_{B\alpha}} \\ & - \sum_{pqrs,B} \langle\alpha|E_{pq,rs}|B\rangle C_B(\beta) \left[ \sum_e \frac{u_{pe} g_{eqrs}}{\epsilon_e - \epsilon_q + \epsilon_r - \epsilon_s + \Delta E_{B\alpha}} \right. \\ & \left. + \sum_e \frac{g_{pers} u_{eq}}{\epsilon_e - \epsilon_q + \Delta E_{B\alpha}} + \frac{1}{2} \sum_{(a,b)} \frac{g_{parb} g_{aqbs}}{\epsilon_a - \epsilon_q + \epsilon_b - \epsilon_s + \Delta E_{B\alpha}} \right] \\ & - \sum_{pqrstu,B} \langle\alpha|E_{pq,rs,tu}|B\rangle C_B(\beta) \sum_e \frac{g_{pers} g_{equ}}{\epsilon_e - \epsilon_q + \epsilon_r - \epsilon_u + \Delta E_{B\alpha}} + (\alpha \leftrightarrow \beta), \end{aligned} \quad (5)$$

where  $u_{pq}$  and  $g_{pqrs}$  are one and two electron integrals, respectively, and  $\Delta E_{B\alpha}$  is the difference between the energies of the zeroth order state and configuration  $E_B^{(0)} - E_\alpha^{(0)}$ . The orbital labels  $\{i, j\}$ ,  $\{a\}$ , and  $\{e\}$  are for doubly occupied, active, and external orbitals, respectively, and  $\{a', b'\}$  run over both active and external orbitals, and the suffix of the generator  $\{p, q, r, s, t, u\}$  run over only active orbitals. The terms including doubly occupied orbitals are omitted in this equation. See Ref. [7] for the full formula. Note that the MRMP energy is also calculated with the formula by setting the number of the states to one.

The formula including doubly occupied orbitals might look tedious. However, the energy can be calculated just a sum of simple terms, hence rather simple. The computation is done with the coupling coefficient driven method. These coupling coefficients are sparse and can be pre-screened according to the condition,

$$\left(v_B^{pq\dots rs}\right)_{\alpha\beta} = \langle\alpha|E_{pq\dots rs}|B\rangle C_B(\beta) > \delta, \quad (6)$$

where  $\delta = 1 \times 10^{-8}$  is usually sufficient to keep the energy accuracy better than  $10^{-5}$  hartree. Thus the multiple summations for active orbitals in Eq. (6), which seemingly scales as the power of the number of active orbitals, are actually diminished considerably.

The computational time strongly depends on the systems calculated: in large CAS cases, the coupling coefficient computation occupies the most of CPU time, and in large basis set cases, integral transformation occupies the time. However, roughly, we can say that MR2D runs in memory, disk, and CPU time comparable to the CAS-SCF computation itself.

MR2D has —not only the option for CAS— also options for general configuration spaces.[9] For general configuration space cases, additional non-CAS terms appear as well as CAS terms, Eq. (6). The non-CAS terms may be calculated through

$$\left(H_{\text{non-CAS}}^{(2)}\right)_{\alpha\beta} = \mathbf{v}^T(\alpha)\mathbf{w}(\beta) \quad (2)$$

with

$$\mathbf{v}_I(\alpha) = \sum_{A \in \text{ref}} \langle I | H | A \rangle C_A(\alpha), \quad (3)$$

$$\mathbf{w}_I(\beta) = \sum_{B \in \text{ref}} \langle I | H | B \rangle C_B(\beta) / (E_\beta^{(0)} - E_B^{(0)}). \quad (4)$$

The intermediate determinants/CSFs  $I$  are constructed by exciting one or two electron(s) from the reference determinants/CSFs within the active orbital space. In general, the number of  $I$  is not large, and thus they may be managed in computer memory.

Analytic gradients are not available in the current version.

### 2.3 TCE – Combination with Tensor Contraction Engine

UTChem also supports a spectrum of single-reference many-electron theories such as configuration-interaction theory (CISD, CISDT, CISDTQ), coupled-cluster theory (CCD, LCCD, CCSD, LCCSD, QCISD, CCSDT, CCSDTQ), and Møller-Plesset perturbation theory (MP2, MP3, MP4). To expedite the time-consuming and error-prone processes of deriving the working equations of these complex theories and implementing efficient computer programs on the basis of these equations, we have taken a unique approach --- we have developed a symbolic manipulation program (Tensor Contraction Engine or TCE)[1] that automated the both processes of the formula derivation and program synthesis. Provided a well-defined ansatz of a second-quantized many-electron theory, TCE performs valid contractions of creation and annihilation operators according to Wick's theorem, consolidates identical terms, and reduces the expressions into the form of tensor contractions acted upon by permutation operators. Subsequently, it determines the order of contractions having the minimal operation and memory cost, factorizes common contractions (defines intermediate tensors), and identifies reusable intermediates. The resulting ordered list of binary tensor contractions, additions, and index permutations is translated into an optimized program that is combined with UTChem. The programs synthesized by TCE take advantage of spin, spatial, and index permutation symmetries simultaneously to minimize the number of arithmetic operations and storage requirement, adjust the peak local memory usage by index range tiling, and support parallel I/O interfaces and dynamic load balancing for parallel executions. TCE not only expedites the daunting task of deriving and implementing various many-electron theories, but also enhances portability, maintainability, and extensibility of the synthesize program, facilitates parallelization and other laborious optimization of the programs which may be tailored to a particular computer architecture, and helps design and tests new many-electron theories which may be too complex to be hand-coded. Currently, the TCE-generated many-electron theories are limited to non-relativistic Hartree-Fock reference wave functions, but the extension to relativistic 2- and 4-component reference wave functions is underway.

## 2.4 LMP2 – An MP2 Method with Localized Molecular Orbitals

Local MP2 (LMP2), which is based on the second-order Møller–Plesset perturbation theory, is an electron correlation method with localized molecular orbitals. Local correlation methods efficiently estimate the dynamical electron correlation in comparison with conventional canonical methods.

Using orthogonal localized occupied orbitals we have developed and implemented the local MP2 method based on the idea developed by Head-Gordon and coworkers. A subset of non-orthogonal correlation functions (the orbital domain) is assigned to each of the localized occupied orbitals using a distance criterion and excitations from localized occupied orbitals that are arranged into subsets. The correlation energy is estimated using a partial diagonalization and an iterative efficient method for solving large-scale linear equations. The orbital domain sizes are found to be independent of the molecular size, and the present local MP2 method covers about 98–99% of the correlation energy of the conventional canonical MP2 method.

Generally, the correlation energy of the second-order Møller-Plesset method is given as

$$E_2 = \frac{1}{4} \bar{\mathbf{V}}^t \bar{\mathbf{T}} \quad (\bar{\mathbf{T}} = \Delta^{-1} \bar{\mathbf{V}}), \quad (5)$$

where the vector  $\bar{\mathbf{T}}$  is the electronic excitation amplitude and vector  $\bar{\mathbf{V}}$  is a vector described by the two-electron repulsion integrals as

$$V_{iajb} = (ia|jb) - (ib|ja). \quad (6)$$

The terms  $i$  and  $j$  denote occupied orbitals, and the terms  $a$  and  $b$  denote the non-orthogonal virtual orbitals. The matrix  $\Delta$  is described by

$$\Delta_{iajb, i'a'j'b'} = (\mathbf{F}_2)_{ia,i'a'} (\mathbf{S}_2)_{jb,j'b'} + (\mathbf{S}_2)_{ia,i'a'} (\mathbf{F}_2)_{jb,j'b'}. \quad (7)$$

Using the overlap and Fock matrices, the matrices  $\mathbf{F}_2$  and  $\mathbf{S}_2$  in Equation (6) are described as follows

$$\begin{aligned} (\mathbf{F}_2)_{ia,i'a'} &\equiv F_{ii'} S_{aa'} - S_{ii'} F_{aa'}, \\ (\mathbf{S}_2)_{ia,i'a'} &\equiv S_{ii'} S_{aa'}. \end{aligned} \quad (11)$$

The non-iterative local MP2 method that was developed by Head-Gordon et al. has to simultaneously diagonalize both the  $\mathbf{F}_2$  and  $\mathbf{S}_2$  matrices to obtain the matrix  $\Delta^{-1}$ . Since the side length of both of these two matrices is the number of occupied orbitals times the number of neighboring virtual orbitals, then these diagonalizations require a high computational cost for large-sized molecules.

In our improved method, this particular step is divided into two steps, which involve the diagonalization of the  $\mathbf{S}_2$  matrix, and the iterative solving of the  $\bar{\mathbf{T}}$  vector without diagonalizing matrix  $\mathbf{F}_2$ . Since  $\mathbf{S}_2$  is a block diagonal matrix within subspaces consisting of same occupied orbitals, then it is easy to diagonalize it. After diagonalizing the  $\mathbf{S}_2$  matrix, the vectors of small norms are excluded to avoid a linear dependence in the virtual space. In the next step, the  $\bar{\mathbf{T}}$  vector is solved by an iterative method without diagonalizing matrix  $\mathbf{F}_2$ .



## 2.5 DFT – Density Functional Theory

Density functional theory (DFT) is a methodology that usually solves the Kohn-Sham (KS) self-consistent field (SCF) equation. In this program, as usual in packages of computational chemistry, the matrix algorithm of the KS method is implemented with a basis set.

The main characteristics of this DFT program are as follows;

1. Johnson's linear-scaling method[10] is implemented.  
Numerical calculations are carried out only for grid points where the adopted Gaussian-type basis functions provide inneglegible values.
2. A screening scheme is carried out for the weight and loop of grids.  
Stratmann-Scuseria-Frisch's weight and microbatch schemes of grid points[11] are also employed. Screenings are also performed for the values of Fock matrix element and electron density.
3. Computational process is optimized for general contraction Gaussian basis functions.
4. OP correlation functional[12,13] and long-range exchange correction (LRXC) scheme[14,15] are implemented.
5. Data of grids are stored in core memory.

In this program, various types of functionals are available besides OP correlation.

## 2.6 REL4D – Relativistic Molecular Theory

One of the main features in the UTChem program package is to be able to consider the relativistic effects. The relativistic effect has been considered as an essential factor to figure out molecular structures, chemical activities, or various properties of heavy-element systems theoretically. Recently many quantum chemists have dedicated a lot of efforts to the calculation and treatment of the electronic structures of polyatomic systems including heavy elements, which are involved in many interesting chemical and physical phenomena. They still present unique difficulties to the theoretical study. Before recent years the relativistic effect had ever been thought less important for chemical properties because the relativity appears primarily in the core electrons, which had been believed to be unlikely to affect chemically active valence regions dramatically. Recent studies, however, have revealed not only quantitatively but also qualitatively that the relativistic effect plays essential and comprehensive roles in total natures of molecular electronic structures for heavy-element systems. We are nowadays convinced that the relativistic effect is definitely important for the accurate theoretical treatment of heavy-element systems as well as the electron correlation effect.

We have recently developed several relativistic molecular theories including four-component relativistic and approximate quasi-relativistic approaches. For the four-component relativistic approach, which is the rigorous relativistic treatment using the four-component spinors explicitly, our development starts from derivation of the analytical form of the basis spinor. In the four-component case we have to manage to get rid of the variational collapse coming from using the finite basis sets. We have proposed the efficient four-component basis spinors using the generally contracted kinetically balanced Gaussian-type spinors, which are taken from the analytical spinor forms corresponding to the one-electron hydrogenic system, and maintain the

variational stability through the kinetic balance condition. Furthermore, our contraction basis spinors have merits to give the correct behavior to the atomic limit, and require significantly less number of small-component basis spinors than the conventional decoupled scalar spin-orbitals used in the pioneering four-component programs MOLDIR and DIRAC. We have formulated and implemented Dirac-Hartree-Fock (DHF)[16] and Dirac-Kohn-Sham (DKS)[17] methods, which have been made computationally feasible by exploiting the highly efficient integral method for two-electron repulsion integrals over the two-component basis spinors. Adoption[18] of the ACE formula proposed by Ishida[3] and the transfer relation proposed by Head-Gordon and Pople[19] make our algorithm efficient. It is shown that our four-component method is the most straightforward and efficient way by comparing with the other four-component relativistic methods and programs.

Even though we have spent much effort to develop the four-component method, still, the relativistic method using four-component spinors demands severe computational efforts to solve, and has a lot of difficulties to be applied to the large-size molecules. Thus, we have been proposed as the quasi-relativistic approximation instead of solving Dirac equation with the fully four-component spinors. Chemists' interest is devoted to the electronic solution, although the four-component calculations provide the eigenstates for not only electronic states but also positronic ones. The motivation of two-component approximation is to obtain the electronic states without treating the positronic solution explicitly. However, since the relativistic effect appearing in the electronic states is closely related to the positronic states in the four-component language, the problem of two-component method is how we can effectively retrieve the effect lost by neglecting the positronic solutions within the framework of the two-component Hamiltonian. In our recent works, we have developed the two kinds of two-component quasi-relativistic method using the RESC[20,21] and higher-order Douglas-Kroll (DK)[22,23] Hamiltonian. The RESC Hamiltonian has been derived from the eliminating small component equation with the relativistic substitutive correction. The higher-order DK Hamiltonian is the effective Hamiltonian obtained by transforming the Dirac Hamiltonian by an exponential-type unitary operator based on the philosophy of DK transformation[24,25] The exponential-type operator, which we have proposed, can simplify the formulation and implementation of the higher-order transformation by taking full advantage of the Baker-Cambell-Hausdorff expansion. Both two-component methods have the excellent features that they contain no singularity, therefore, and variationally treat the relativistic effect on the same footing as the electron correlation.

The resulting four-component relativistic and approximate quasi-relativistic codes have been packed in a program suite named REL4D, which is currently a part of the UTChem program package. The efficient ERI engine contributes to the performance for generating relativistic molecular integrals. In the self-consistent field (SCF) routine, several techniques have been implemented; the restricted open-shell treatment proposed by Davidson, the direct and conventional SCF methods, and the direct inversion in the iterative subspace (DIIS) method. The numerical grid-quadrature routine for the exchange-correlation potentials is implemented for our DKS scheme. The (SSISS) integrals can be treated near the SCF convergence in the direct SCF approach.

## 2.7 S&D – Simulation and Dynamics

*S&D* is a program for executing molecular dynamics and Monte Carlo simulations. *Ab initio* molecular dynamics and Monte Carlo simulations can be carried out. Molecular dynamics simulations are carried out in microcanonical (NVE) ensemble. Verlet's algorithm and Gear's algorithm can be selected for integrating the equation of motion. *Ab initio* molecular dynamics can be performed with any approach in UTChem, in which the energy gradients are available. Monte Carlo simulations are performed in canonical (NVT) ensemble. In Monte Carlo simulation using *ab initio* energies, the energies are calculated in other programs. Simulated annealing and the replica exchange method (REM)[26,27,28,29] are also available for both classical and *ab initio* Monte Carlo simulations. In REM, a number of non-interacting replicas of the original system at different temperatures are simulated independently and simultaneously. REM is a powerful tool to obtain the global minimum in systems with many local minima, and difficult to optimize. Periodic boundary condition or spherical (image) boundary condition can be selected in the simulations.

The Quantum Mechanical (QM) / Molecular Mechanical (MM) method is available in UTChem. Simulation using QM/MM energies and gradients can be performed as well. QM/MM calculation can be carried out in Hartree-Fock methods and density functional theories. Geometry optimization using QM/MM is not available in the current version.

**Acknowledgments.** This research was supported in part by a grant-in-aid for Scientific Research in Specially Promoted Research "Simulations and Dynamics for Real Systems" from the Ministry of Education, Science, Culture, and Sports of Japan, and by a grant from the Genesis Research Institute. Tensor Contraction Engine is a product of Battelle and Pacific Northwest National Laboratory, and its development has been funded by the U.S. Department of Energy, the Division of Basic Energy Science, Office of Science under contract DE-AC06-76RLO 1830 with Battelle.

## References

1. Webpage of the Tensor Contraction Engine (TCE) project is placed on <http://www.cis.ohio-state.edu/~gb/TCE/>.
2. T. H. Dunning Jr., J. Chem. Phys. 90, 1007 (1989).
3. J. Almlöf and P. R. Taylor, J. Chem. Phys. 86, 4070 (1987).
4. K. Ishida, Int. J. Quantum Chem. 59, 209 (1996); K. Ishida, J. Chem. Phys. 109, 881 (1998); K. Ishida, J. Comput. Chem. 19, 923 (1998); K. Ishida, J. Chem. Phys. 111, 4913 (1999).
5. T. Yanai, K. Ishida, H. Nakano, and K. Hirao, Int. J. Quantum Chem. 76, 396, 2000.
6. Global Array (GA) toolkit developed in Pacific Northwest National Laboratory, <http://www.emsl.pnl.gov:2080/docs/global>.
7. H. Nakano, J. Chem. Phys. 99, 7983 (1993); H. Nakano: Chem. Phys. Lett. 207, 372 (1993).
8. K. Hirao, Chem. Phys. Lett. 190,374 (1992); K. Hirao, Chem. Phys. Lett. 196, 397 (1992); K. Hirao, Intern. J. Quantum Chem. Symp. 26,517 (1992).

9. H. Nakano, J. Nakatani, and K. Hirao, *J. Chem. Phys.* 114,1133 (2001); H. Nakano, R. Uchiyama, and K. Hirao, *J. Comput. Chem.* 23, 1166 (2002).
10. C. A. White, B. G. Johnson, P. M. W. Gill and M. HeadGordon, *Chem. Phys. Lett.* 253, 268 (1996).
11. R. E. Stratmann, G. E. Scuseria and M. J. Frisch, *Chem. Phys. Lett.* 257,213 (1996).
12. T. Tsuneda, T. Suzumura and K. Hirao, *J. Chem. Phys.* 110, 10664 (1999).
13. T. Tsuneda, T. Suzumura and K. Hirao, *J. Chem. Phys.*, 111, 5656 (1999).
14. H. Iikura, T. Tsuneda, T. Yanai, and K. Hirao, *J. Chem. Phys.* 115,3540 (2001).
15. M. Kamiya, T. Tsuneda and K. Hirao, *J. Chem. Phys.*,117, 6010 (2002).
16. T. Yanai, T. Nakajima, Y. Ishikawa, and K. Hirao, *J. Chem. Phys.* 114, 6526 (2001).
17. T. Yanai, H. Iikura, T. Nakajima, Y. Ishikawa, and K. Hirao, *J. Chem. Phys.* 115, 8267 (2001).
18. T. Yanai, T. Nakajima, Y. Ishikawa, and K. Hirao, *J. Chem. Phys.* 116, 10122 (2002).
19. M. Head-Gordon and J. A. Pople, *J. Chem. Phys.* 89, 5777 (1988).
20. T. Nakajima and K. Hirao, *Chem. Phys. Lett.* 302, 383 (1999).
21. T. Nakajima and K. Hirao, *J. Chem. Phys.* 113, 7786 (2000).
22. T. Nakajima, T. Suzumura, and K. Hirao, *Chem. Phys. Lett.* 304, 271 (1999).
23. T. Nakajima and K. Hirao, *Chem. Phys. Lett.* 329, 511 (2000).
24. M. Douglas and N. M. Kroll, *Ann. Phys. (N.Y.)* 82, 89 (1974).
25. B. A. Hess, *Phys. Rev. A* 33, 3742 (1986).
26. K. Hukushima and K. Nemoto, *J. Phys. Soc. Jpn.* 65, 1604 (1996).
27. C. J. Geyer and E. M. Keramidas (Ed.), *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, Interface Foundation, Fairfax Station 156 (1991).
28. R. H. Swendsen and J.-S. Wang, *Phys. Rev. Lett.* 57, 2607 (1986).
29. Y. Ishikawa, Y. Sugita, T. Nishikawa, Y. Okamoto, *Chem. Phys. Lett.* 333, 199 (2001).