UDC 530.145: 681.3.06

Examples of Using the VPP500 in JRCAT

●Takahiro Yamasaki ●Koichi Sato ●Kiyoyuki Terakura

(Manuscript received May 12,1997)

We have been using calculation techniques based on first-principles (or quantum) molecular dynamics to analyze physical phenomena at the atomic level. Some examples of the phenomena that have been analyzed are molecule dissociative adsorption reactions on semiconductor and metal surfaces, and metal-insulator phase transitions of organic solids. It is certain that such analyses will be more widely and frequently performed in the future.

The VPP500 vector parallel computer is very compatible with these calculation techniques. When programs written for vector computers are run on the VPP500, the parallel acceleration rate can be improved considerably without reducing the vectorization factor. The VPP500 can handle large-scale calculations because it separates calculations from data and allows calculations to be assigned to processor elements appropriately.

This paper describes our experience of changing first-principles molecular dynamics programs so that they can be parallel-executed on the VPP500, and describes the operation management of a computer group centered around the VPP500. This paper also looks at some simulation results.

1. Introduction

Supercomputers play a major role in investigating solid-state properties. The first principles calculations with a supercomputer[Note)] are used to explore phenomena that could not be experimentally observed or understood because they occur in a microscopic area of space and time. One of the fields where such calculations are frequently performed is the molecular reaction and growth on semiconductor surfaces. The number of simulations performed using first principles calculations will certainly increase in the future.

The Joint Research Center for Atom Technology (JRCAT), to which the authors belong,1) is a research organization based on a joint study agreement for equal-partnership research between the Agency of Industrial Science and Technology, the National Institute for Advanced Interdisciplinary Research (NAIR), and Angstorm Technology Partnership (ATP). The major objective of the organization is to advance materials science by preparing new atomically-ordered structures or ultramicron (angstrom-scale) structures in or on a solid. JRCAT consists of eight experimental groups, one theoretical group, and the system operation department, which controls and maintains the computer environment. The theoretical group consists of three subgroups to cover semiconductors, transition metal compounds, and exotic materials, respectively.

Since 1994 we have been using computers such as the VPP500, CM5E, DEC10000, and IRIS ONYX to perform first principles molecular dynamics calculations. This paper describes our experiences with these computers, especially with program parallelization in the VPP500 and the operation methods. Our experiences will also be of interest to VPP300/700 series users because source code developed on the VPP500 is compatible with those machines.

Note) Calculation based on the quantum theory without using experimental values obtained from experimental observations.

2. Calculation Examples

First, we will describe some examples of the calculations performed using the VPP500.

2.1 Type-C defect structure on Si(001) surface[2)]

Defects exist on all silicon surfaces, and the typical defects are classified into types A, B, and C. Surface silicon atoms form dimers to become energetically stabilized. A recombined double silicon atom is called a dimer. The type A defect is a defect of one surface dimer, and the type B defect is a defect of two surface dimers. The type C defect is important from a technological point of view, but studies have not yet revealed its structure. In the oxidation process, the type C defect is the most sensitive reaction site with oxygen molecules. The simple structure of the type C defect as revealed by a scanning tunneling microscope (STM) was used as an initial geometry and then optimized using first principles molecular dynamics. The structure obtained had a completely different structure from the one seen in the STM image. However, relaxing a structure with one second-layer atom removed reproduced the experimental STM image (**Fig. 1**). The unit cell of the system includes 390 atoms.

2.2 Initial homoepitaxial growth process on Si (001) surface[3)]

Experiments have proved that molecular beam epitaxy (MBE) on an Si (001) surface allows the formation of long dimer rows growing vertically to the substrate dimer rows at a substrate temperature of around 500 K. They have also revealed other interesting phenomena, for example, the disappearance at high temperature of a short, dilute dimer row observed at low temperature. We optimized a structure and obtained its energy with various kinds of surface adsorption clusters of Si atoms. Then we calculated geometrical paths in several important growth processes (**Fig. 2**) and also evaluated the activation energies. The results enables us to explain the change in growth mechanisms over different temperatures.

2.3 Early oxidation process on Si (001) surface[4)]

Despite the importance of silicon surface oxidation in device manufacturing, few studies have theoretically analyzed the oxidation using the first principles methods. Obstacles to a successful theoretical analysis include the consideration of spins and an inadequate local density approximation (LDA) of the electronic density function. (Taking spins and general gradient approximation into consideration increases the computation amount.) Because not all of the possible reactions can be traced, a structure in which oxygen molecules are arranged in parallel was used as an initial geometry, and four reaction processes were traced. **Figure 3** shows one of the processes. Only one process reacted without an energy barrier when the spin flip energy on the silicon surface was disregarded. The actual reaction process requires the spin flip energy, which may correspond to the energy barriers observed. This study is continuing in other areas, for example, the oxidation process for the type-C defect.

2.4 Si cluster[5)]

The Kanayama group of JRCAT is trying to use captured and grown cluster ions as seeds to prepare nanometer structures. The group grows hydrogenated silicon cluster ions made from monosilane gas. Experiments have yielded a possible proportion of hydrogen to silicon, but not the ion structure. Many hydrogenated silicon cluster ion structures have been optimized to determine their stable structures and their energy details. **Figure 4** shows the stable structure of Si_6 and its electronic state.

2.5 Organic conductive material (R_1-R_2-DCNQI)$_2$M[6)]

The organic conductive material (R1-R2-DCNQI)$_2$M has a complex structure (**Fig. 5**). CH_3, I, and Br are used as the substituents of R_1 and R_2; and Li, Cu, and Ag are used as the metallic element of M. The solid-state properties greatly

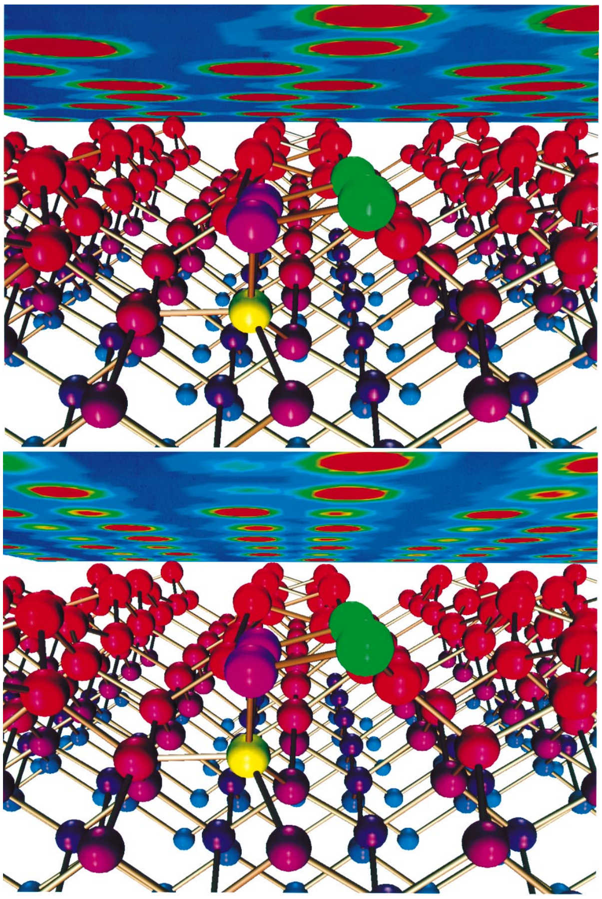


Fig.1— C-type defect on Si (001) surface.

Structural optimization was performed for a structure where a Si atom on the second surface was removed. The yellow atom, which belonged to the third surface, recombines with the neighboring atom. The surface atoms (pink) are pulled down. The top and bottom figures overlap with cross-sections of electronic density of states in energy ranges of Ef (Fermi energy level)-1 to Ef and Ef to Ef+1eV, respectively. The top figure shows the STM occupied state, and the bottom figure shows the STM unoccupied state.

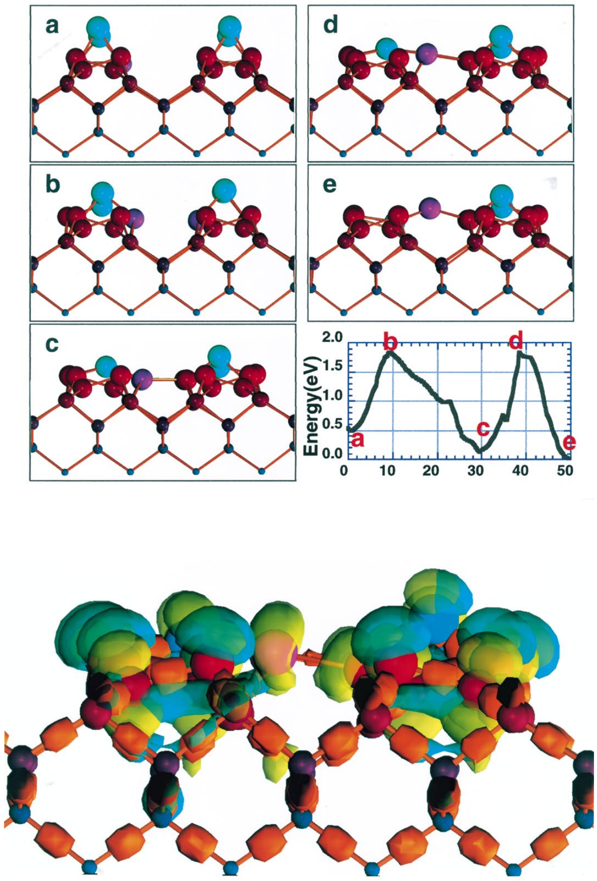


Fig.2— Si-dimer condensation process on Si (001) surface.

The light-blue atoms are adsorbed dimers. The graph shows the change in energy when the pink atoms are forcefully moved. The horizontal axis represents atom displacement (atom unit). The structure in e, with densely arranged adsorbed dimers, is energetically more stable than structure a, which has loosely arranged absorbed dimers. Interestingly, there is another structure, structure c, which is between structures a and e and is more stable than structure a. The bottom figure shows the electronic state of structure d. The orange parts show the combined electrical charge. The yellow and light-blue clouds show the spatial distribution in the energy ranges of Ef to Ef-1 and EF to Ef+1eV, respectively. The part where yellow and light-blue clouds overlap is in the metallic electronic state.

depend on R1 and R2 when Cu is used as M. The DCNQI (Dicyanoquinodiimine) molecule has a planner structure and forms a one-dimensional chain. The chains are three-dimensionally connected with each other through M. This material has attracted attention because the solid-state properties may be controlled by changing R_1, R_2 and M. With CH_3 for R_1 and R_2, we observed a metal-insulator-metal transition when the temperature was lowered under slight pressure. When I was used as a substituent, the system remained as a metal even at low temperature under normal pressure. When Br was used as a substituent, the system underwent metal-insulator transition.

Electronic state calculations have shown that the specificity of Cu as M results from strong hybridization between the unoccupied orbital of the DCNQI molecule and the d orbital of Cu. Such hybridization may increase three-dimensionality,

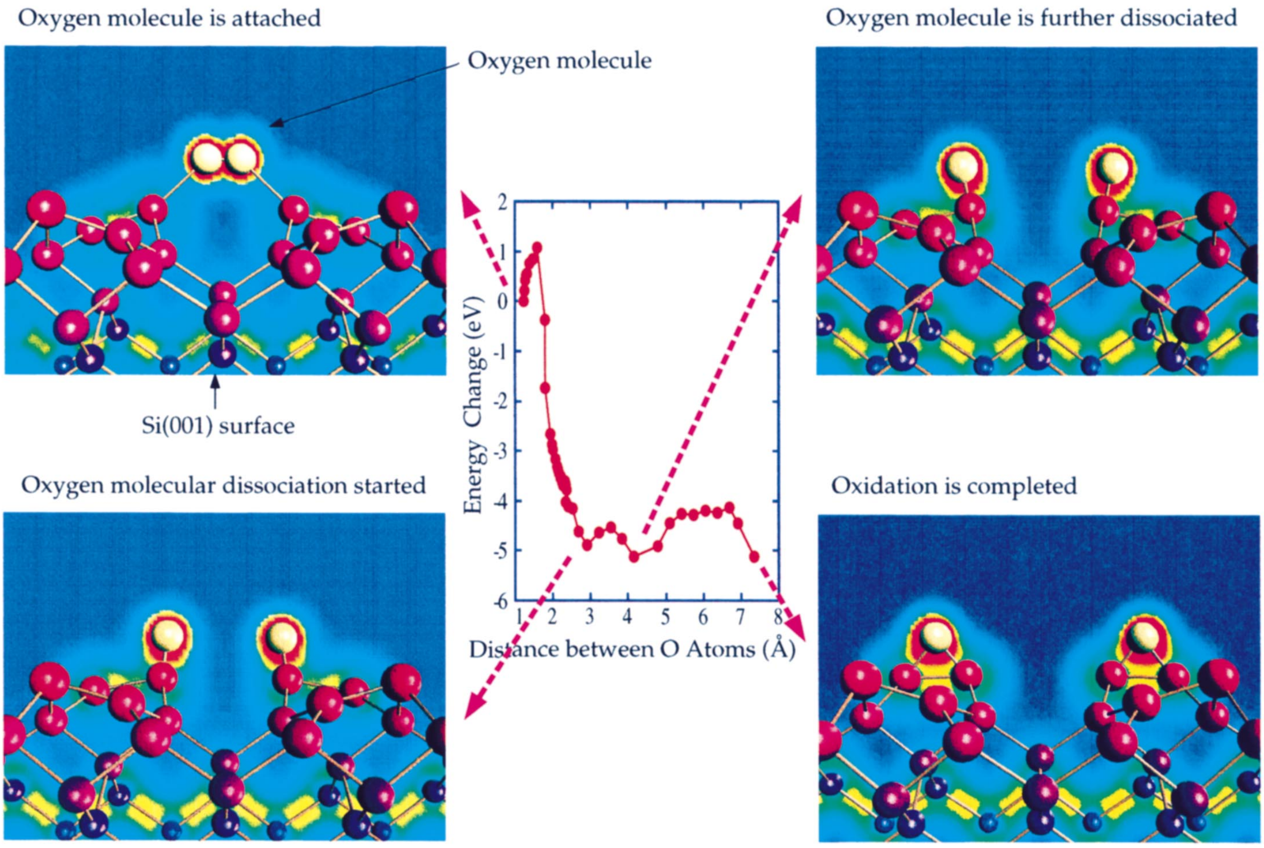


Fig.3— Process of O_2 molecule chemisorption on Si (001) surface.

Reaction proceeds counterclockwise from top-left figure. The white and red balls are oxygen and silicon atoms, respectively. The figures also show the surface charge density distributions at the points where the oxygen molecules are dissociated. The central graph shows the relation between the distance between oxygen atoms and the energy change. There is an energy barrier of about 1 eV.

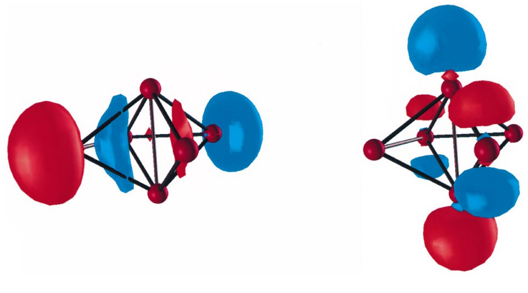


Fig.4— Si6 Cluster.

The highest occupied molecular orbital (HOMO, left) and lowest unoccupied molecular orbital (LUMO, right) are shown on the clusters. The color of the clouds changes with the phase of the wave function. The blue and red clouds represent negative and positive areas, respectively.

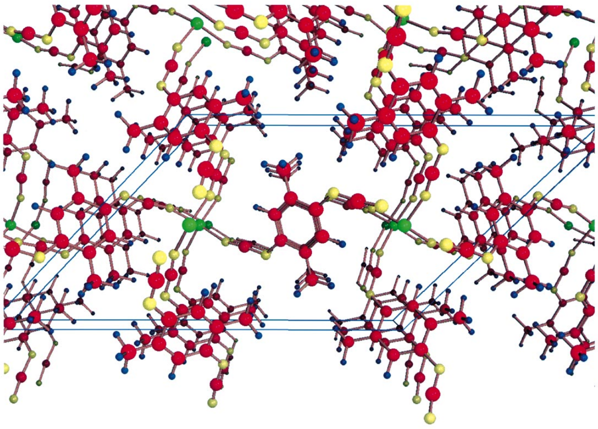


Fig.5— (DMe DCNQI)2 Cu.

The red, yellow, blue, and green atoms are carbon, nitrogen, hydrogen, and copper, respectively. The DCNQI molecule forms one-dimensional chains perpendicular to the figure, forming three-dimensional structures through the copper atoms.

which allows some DCNQI molecules to remain as a metal at low temperature. When the substituent is CH_3, a charge-density wave of periodicity 3c is likely to occur along the c axis, which could explain the change in the stable phase under slight pressure.

3. Parallelization of the First Principles Molecular Dynamics

This section describes some calculation and parallelization methods.

3.1 Car-Parrinello method

To determine the electronic state of a material, the eigenvalue equation obtained by the one-body approximation according to the density functional theory is solved to determine the wave function. The equation has to be solved repeatedly until the input and output of the charge density become self consistent (**Fig. 6**). The total energy of a system with its electronic state determined using the above method was varied for ion coordinates to determine the force on ions.

The conventional method takes the matrix diagonalization to determine the eigenvalues and the eigen vectors (i.e., wave functions). The number of operations required to determine the eigenvalues and the eigen vectors is proportional to N^3 and the storage area required is proportional to N^2 when the number of the basis functions in a wave function is N. Therefore, the calculation for a large system with 1,000 or more atoms has been impracticable.

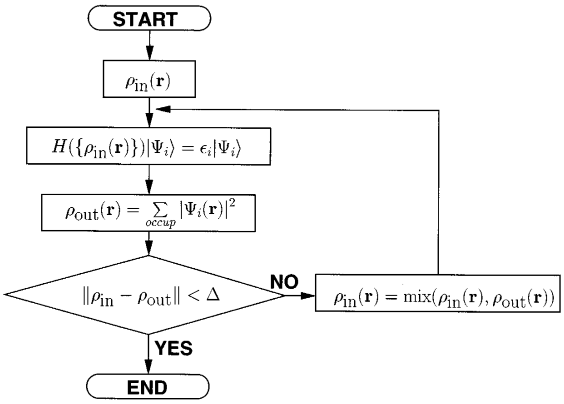


Fig.6— Algorithm of electronic structure calculation.

1) An initial charge density is given, then the potential is calculated.
2) The Schrödinger equation is solved to determine the characteristic vector (wave function).
3) The wave function is used to construct the charge density.
4) The input and output charge densities are compared. The calculation is completed when the densities are considered to converge.
5) If the convergence is insufficient, a new input charge density distribution is constructed to return to the start.

Lagrangian

$$\begin{aligned} L &= \mu \sum_{i=1}^{M} \langle \dot{\Psi}_i | \dot{\Psi}_i \rangle + \frac{1}{2} \sum_{n=1}^{N_I} M_n |\dot{\vec{R}}_n|^2 \\ &- E[\{\Psi_i\}, \{\vec{R}_n\}] + \sum_{i,j=1}^{M} \lambda_{j,i} (\langle \Psi_i | \Psi_j \rangle - \delta_{i,j}) \end{aligned}$$

Euler-Lagrange's Equations of motion

$$\begin{aligned} \mu |\ddot{\Psi}_i\rangle &= -H|\Psi_i\rangle + \sum_{j=1}^{M} \lambda_{j,i} |\Psi_j\rangle \\ M_n \ddot{\vec{R}}_n &= -\nabla_n E[\{\Psi_i\}, \{\vec{R}_n\}] \end{aligned}$$

Fig.7— The Car-Parrinello method.

The method developed by Car and Parrinello in 1985⁷⁾ starts with the Lagrangian and introduces the equations of motion for wave functions and ions to develop them (**Fig. 7**). The electron temperature is gradually reduced to determine the most stable electronic state of the system. In practice, the Gram-Schmidt orthonormalization method is combined with the more efficient steepest descent method, conjugate gradient method, or residual minimization method to update a wave function. The number of operations to determine a wave function with an equation of motion falls to M (number of eigenstates) $\times N \times N$ (in general, M is much smaller than N). The pseudopotential method allows us to deal with valence electron states only. Moreover, expanding the wave function with plane waves enables us to use a fast Fourier transform, which reduces the number of operations to $M \times N \times \log N$. This method also reduces the required storage area to $M \times N$. These techniques have allowed us to perform the first principles molecular dynamics calculations of large systems, which were previously impossible.

A three-dimensional mixed-radix FFT must be efficient because the FFT consumes most of the CPU time. An FFT box that covers the space of the plane-wave vectors used to expand the wave functions should be selected. The mixed-radix FFT is required to minimize the waste area in the FFT box. Wave function transformation from reciprocal to real-space representation most frequently uses an FFT. This operation is necessary when the product of a potential and wave function is prepared and when the charge density is constructed.

The number of operations needed to orthonormalize wave functions with the Gram-Schdmit method is proportional to $M \times N \times N$. However, the time required for these operations increases with the system size. It is difficult to efficiently parallelize the procedure of the Gram-Schmidt orthonormalization.

3.2 Parallelization strategy

We formed a parallelization strategy after considering the VPP500's performance and the vectorization level of our first principles molecular dynamics program. The strategy is described later. Our main requirement is efficient parallelization for large-system calculations. For large-system calculations, data partitioning is as important as the procedure (operation) partitioning.

1) Vectorization rate improvement

Good vectorization is essential. An ordinary vectorization method can be used without modification.

2) Inter-PE data transfer optimization

Although the network that transfers data between the VPP's processor elements (PEs) has a wide bandwidth, it is not efficient for frequent transfers of small-unit data. The following method however is effective:

- Data transfers of multiple variables and arrays are combined using a work array.
- The inter-PE data transmissions inside a multiple DO loop are performed as much as possible with outside DO loops using a work array or other means.

3) Procedure and data partition unit determination

Four partition axes were prepared.

- Procedures and data were partitioned for the electronic eigenstates because the number of them reached values of from 100 to 10,000 (Partition by energy level). This method allows partitioning with a balanced particle size for each PE. This method also allows each PE to locally perform multiple FFT operations at the same time. An FFT operation program that is difficult to parallelize can be used without parallelization if sufficient memory is locally available to perform such FFT operations. The FFT operation itself must be parallelized for larger systems.
- A coefficient space is partitioned for charge density expansion and related arrays. Although this method mainly aims to compress the storage area, it is also useful for procedure partitioning when the vector length exceeds 2,000 after a procedure is partitioned.
- The number of atoms is used as a partition axis for the force on atoms.
- FFT work arrays are partitioned. Although FFT operations are performed locally, an operation such as an inner product, may be performed on the FFT results. In such a case, multiple FFT arrays are partitioned and stored because they cannot be kept on a single PE. The three-dimensional mixed-radix FFT program developed by JRCAT is incorporated into our Car-Parrinello program.

4) Programs shared with other computers

One set of source files should be shared among multiple computers. Otherwise, multiple programs with the same source must be controlled on different computers, which is inefficient when adding a function or detecting and correcting a bug. Therefore, one set of the original source file is prepared so that it can generate the source file for various models using a C compiler preprocessor.

3.3 Parallelization Procedure

Our parallelization goal was for 20 PEs to achieve 10 times the performance of a single PE. To achieve this goal, we extracted the subroutines which consume 95% of the total CPU time. Cost analysis was done for a relatively small system (a system that can be analyzed with a single PE), and the calculated cost was used to select N subroutines (N subroutines occupy 95% of the total cost). A list of arrays that must be partitioned was then prepared. Usually, the data partition design is prepared, then the program is corrected according to the design. However, such a procedure needs great patience and care because there are several thousand steps in parallelizing the program. Not only the procedure partitioning, but also the data partitioning must be considered for distributed memory computers such as the VPP500. The procedure partitioning tends to have a big effect yet requires a relatively small amount of work, while the data partitioning tends to be complex because data is often used globally throughout the program. Also, adjusting data partitioning methods among subroutines is difficult.

Therefore, we divided the parallelization into two steps. The first step defines for each target subroutine a global array into which the contents of the duplicated local array to be finally eliminated are copied. The global array is used for the procedure partitioning. Then, the results are transferred from the global array to the duplicated local array. At this time, data is unified among all the PEs. In the first step, the duplicated local array is used to transfer data between subroutines. This procedure is completed for all the target subroutines before proceeding to the second step. The second step erases the duplicated local arrays and uses the global arrays to transfer data (**Fig. 8**).

The parallelization was divided into two steps because:

- The parallelization benefit can be realized from an early stage, and
- work can be shared between many people because the first step is conducted for each subroutine independently.

3.4 Gram-Schmidt orthonormalization

The Gram-Schmidt orthonormalization is the most difficult part to parallelize. When wave functions are partitioned according to the energy levels, parallelization by the original algorithm of Gram-Schmidt orthonormalization requires 2M data transfers, while a modified Gram-Schmidt method requires M broadcast transfers only (**Fig. 9**). Moreover, for the wave function partitioning, a cyclic partitioning provides a better particle size

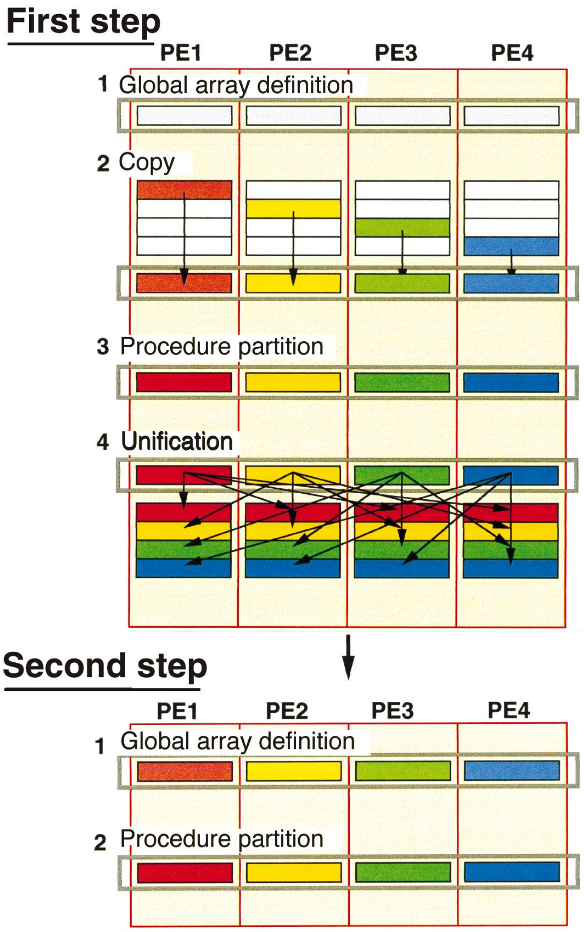


Fig.8— Two-step parallelization.

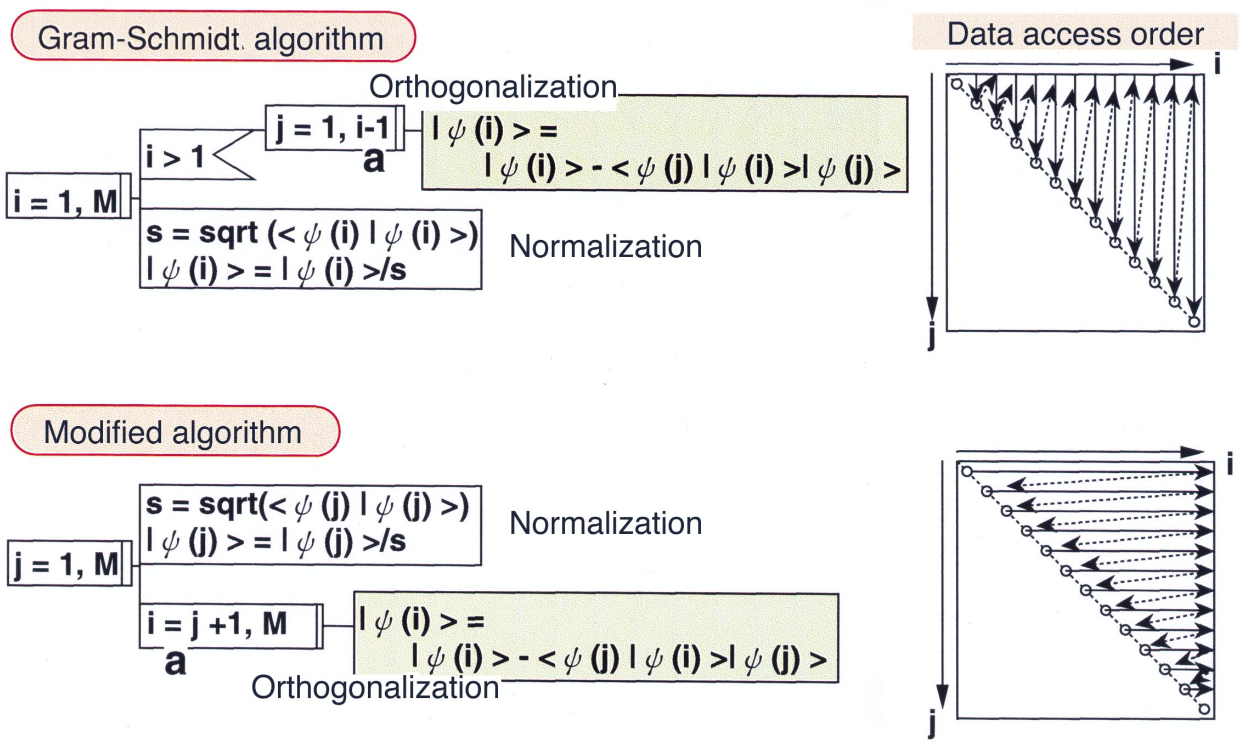


Fig.9— Gram-Schmidt orthonormalization.

The upper part shows the algorithm that closely realizes the original Gram-Schmidt orthogonalization algorithm. The letter M represents the number of wave functions. The two algorithms are different in terms of the order of orthogonalization and normalization, and have opposite orders of duplicate DO loops. When the wave function is partitioned, the original algorithm provides two types of network transfers by parallelizing the DO loop of "a". The two types are broadcast of ø (i) and global sum of < ø (j) | ø (i) > | ø (j) >. On the other hand, the modified algorithm requires only one type of network data transfer or broadcast transfer of ø (i), even when the DO loop of "a" is partitioned.

balance and a higher parallelization performance than a uniform partitioning. This method is beneficial when memory is limited because it does not need additional data arrays.

Wave functions can also be partitioned according to expansion coefficients instead of the energy levels. This method partitions the inner product operation between wave functions. A vector will be effective and provide a good particle size balance if it is sufficiently long after being partitioned. Since partitioning by energy level is more favorable for other subroutines, wave function arrays with two partition axes and data transfer between such arrays are required. This method is useful because of the high parallelization performance when there is sufficient memory. These two methods are used on a case by case basis. In either method, wave functions must be arranged in ascending order of energy eigenvalues. Since arranging them is time-consuming, index vectors that maintain the order are introduced.

3.5 Effect of parallelization

A trial calculation was performed with the parallelized program for the electronic state of the surface structure of diamond. This system has 160 atoms per unit cell. The number of PEs was 1, 8, 16, or 24. **Figure 10** shows how parallelization improves the total performance and the performance of the three subroutines that consume most of the CPU time. The performance with one PE is set to 1. This figure shows good performances for all subroutines except for the one responsible for the Gram-schdmit orthonormalization method. The overall performance with 24 PEs is 21.5 times faster than that with a single PE.

The simulation with eight PEs for a system containing 162 Si atoms per unit cell showed a

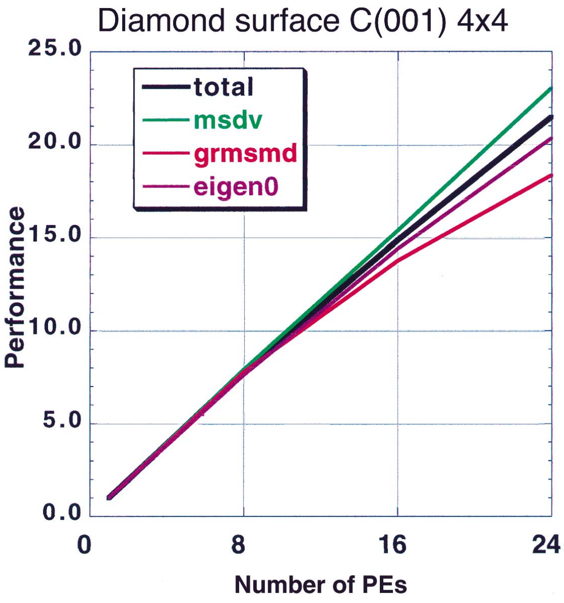


Fig.10— Effect of parallelization on performance.

The graph is normalized with the performance efficiency of one PE set to 1.

total : includes the operations in all subroutines.
msdv : subroutine that updates wave functions
grmsmd : subroutine responsible for the Gram-Schdmit orthonormalization
eigen0 : subroutine that calculates eigenvalues

performance of 745 MFLOPS per PE, which is 45% of the cataloged peak performance.

3.6 Future program development

As described above, the current program has, to an extent, simultaneously improved the parallelization performance and vectorization rate. However, program optimization has almost reached its limit in terms of operation speed; therefore, we are investigating various calculation methods of obtaining optimal solutions in the shortest time. We are examining the effect of conjugate gradient and quasi Newtonian methods on optimizations of wave functions, charge density, and atomic configurations.

4. System Management and Operation

4.1 VPP500 configuration and performance

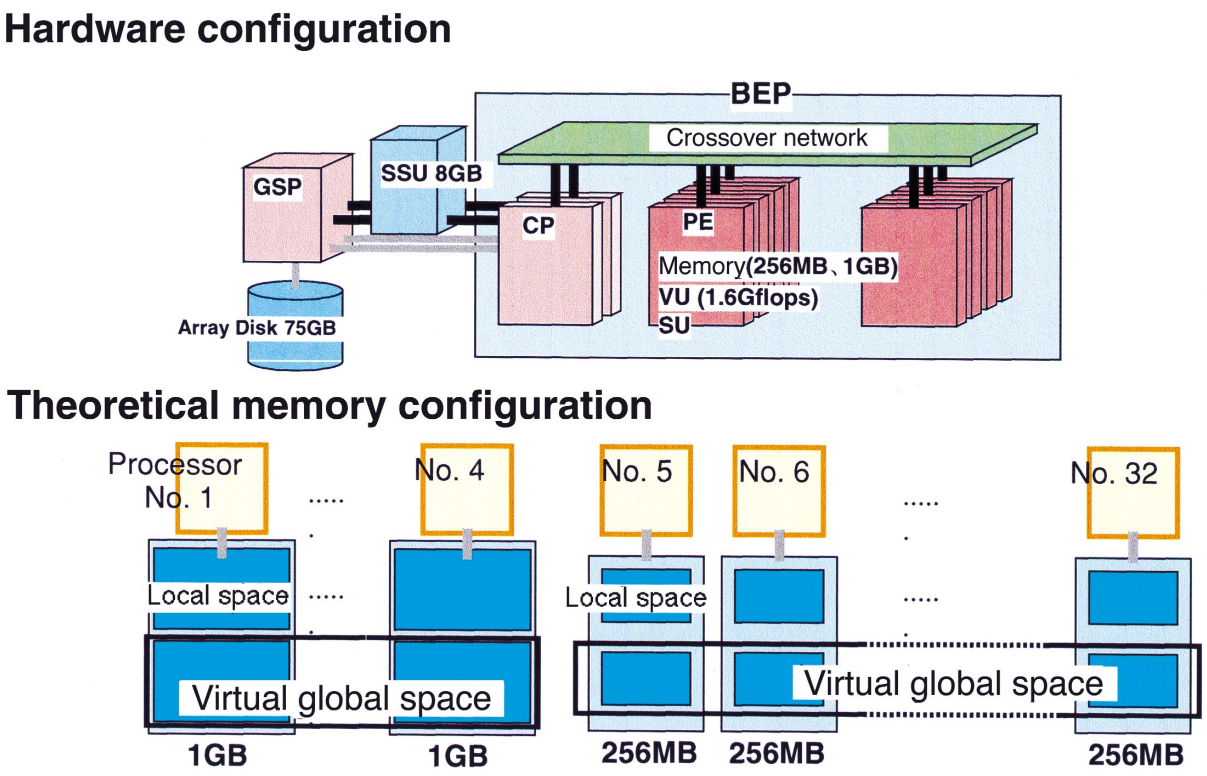
The VPP500 used in JRCAT (**Fig. 11**) has a front-end global system processor (GSP) and a back-end processor (BEP). Users compile files and input jobs at the GSP. The BEP consists of two control processors (CPs) and 32 processor elements (PEs). Each PE consists of a data transfer unit, a 256-megabyte or 1 gigabyte memory unit, a scalar unit, and a vector unit with a cataloged peak performance of 1.6 GFLOPS. FORTRAN77/VPP assumes that there is an intrinsic local memory space in each PE and a global memory space common to all processors. However, this global memory is virtual and is physically included in the memory of each PE. Each PE connects with the other PEs through a crossbar network, which allows simultaneous inter-PE communications. Each PE accesses the global memory through communications. The inter-PE data transfer bandwidth is 400 Mbytes/second. Memory is extended to 1 gigabyte in four of the 32 PEs, and the remaining 28 PEs have a memory of 256 megabytes. In practice, this configuration is equivalent to a system consisting of a GSP (front-end-computer) and two back-end parallel computers, one of which has 4 PEs and the other 28 PEs.

In the most efficient case, the vector operation performance of the VPP500 is almost near to its maximum of 1.6 GFLOPS. The processing speed improvement reaches the maximum in many vector operations when the vector length exceeds 2,000.

We measured the data transfer time with four PEs for the global sum operation, which equalizes all the PE data by addition and is the most frequently used inter-PE data-transfer operations. The result showed that transfer latency is about 40 microseconds. The communication bandwidth of about 400 megabytes/second is good for transferring large amounts of data, but is not efficient for frequently transferring data packets of less than 1 kbyte.

4.2 Overall system configuration

The system connects SUN, HP, and other workstations (including the VPP500/32 as a vec-



Hardware configuration

Theoretical memory configuration

Fig.11— VPP500 of JRCAT.

tor parallel computer); a CM-5E of 128 nodes; a DEC10000 file server, and IRIS ONYX (for visualization) through a network based on the FDDI ring.

Users use the DEC10000 disk area (240 gigabytes) mounted on NFS as a home directory in most of the computers controlled by NIS. Users can access files in the disk area from the GSP and BEP of the VPP. The VPP has another disk area of 75 gigabytes. This area is used for preparing large capacity files because it can be quickly accessed from the BEP using an SSU as a cache. Users can prepare only transient files in the local disk areas of computers other than the DEC and GSP. In principle, users can log in from any computer and use the same file system under the home directory. Users' files are automatically backed up every day. A newly introduced tape robot has enabled us to restore any file that has been deleted within the last year.

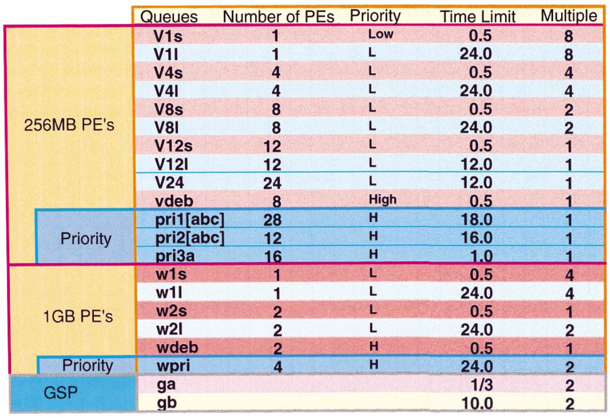
The usage environment has been unified as much as possible. For example, a SUN workstation with the SUN operating system mounts files under /usr/local of an NIS server by using the network file system (NFS).

Users can log in our file system from any workstation. User's home directories are allocated to DEC disk spaces that are optimal for each user and minimize the risk of data loss and the loads on users and computer administrators.

4.3 VPP job control

Figure 12 shows the job queue composition of the VPP500 at JRCAT. The composition has been optimized in previous job inputs. Users who want to complete a job quickly can be given priority.

However, the combination of uncontrolled long-term 1-PE jobs and multiple-PE jobs would reduce PE usage efficiency and increase the waiting time. Therefore, to prevent such a combination, a job swap function was introduced. The function gives priority to a job that was input to the priority queue by swapping out a job in a non-priority queue. Nonetheless, there is still a possi-



		Queues	Number of PEs	Priority	Time Limit	Multiple
256MB PE's		V1s	1	Low	0.5	8
		V1l	1	L	24.0	8
		V4s	4	L	0.5	4
		V4l	4	L	24.0	4
		V8s	8	L	0.5	2
		V8l	8	L	24.0	2
		V12s	12	L	0.5	1
		V12l	12	L	12.0	1
		V24	24	L	12.0	1
		vdeb	8	High	0.5	1
	Priority	pri1[abc]	28	H	18.0	1
		pri2[abc]	12	H	16.0	1
		pri3a	16	H	1.0	1
1GB PE's		w1s	1	L	0.5	4
		w1l	1	L	24.0	4
		w2s	2	L	0.5	1
		w2l	2	L	24.0	2
		wdeb	2	H	0.5	1
	Priority	wpri	4	H	24.0	2
GSP		ga			1/3	2
		gb			10.0	2

Fig.12— Composition of job queues.

bility to wait for a long time because once a job is swapped out, it must go back to the PEs from which it was swapped out. A function to bring a swapped-out job to an empty PE must be carefully examined before its introduction because swapping may take several minutes.

The CPU operating efficiency, which was low in the beginning, has been increasing satisfactorily and sometimes reaches about 90%. This level is fairly good considering that there are many jobs of varying scale being run at the same time.

5. Conclusion

The Car-Parrinello method is well suited to vector parallelized computers. It can use conventional methods to enhance vectorization efficiency and provide a high parallelization performance. One of the reasons for the suitability is the large memory space in each PE, which allows a large procedure partition and reduces the frequency of data transfers between networks. However, no system can currently satisfy the need for quicker simulation of larger and more complex systems. The only way to satisfy this need is to maximize the performance of each PE and then increase the number of PEs. It is therefore necessary to develop parallelization support tools for solving the memory partition loss problems that occur when the number of PEs is increased. It is also necessary to expand the parallelization scientific computation libraries.

This paper has described the results of a study supported by the New Energy Development Organization (NEDO).

References

1) http: //www.jrcat.or.jp
2) T. Uda. and K. Terakura: Structure model for the type-C defect of the Si (001) surface. Physical Review B,**53**, 11, pp.6999-7001 (1996)
3) T. Yamasaki, T. Uda, and K. Terakura: Initial process of Si homoepitaxial growth on Si (001). Physical Review Letters, **76**, pp.2949-2952 (1996)
4) T. Miyazaki et al.: Hydrogenation effects on structures of Silicon clusters. Mat. Res. Soc. Symp. Proc. 408, 1996, pp.553-538.
5) K. Kato et al. : Theory of initial oxidation stages on Si (001) surfaces by spin-polarized generalized gradient calculation, Mat. Res. Soc. Symp. Proc. 408, 1996, pp.427-432.
6) T. Miyazaki et al. : First principles theoretical study of metallic states of DCNQI (Cu, Ag) systems: Simplicity and variety in complex systems. Physical Review Letters, **74**, 25, pp.5104-5107 (1995).
7) R. Car and M. Parrinello : Unified approach for molecular dynamics and density-functional theory. Physical Review Letters, **55**, 22, pp.2471-2474 (1985).

Takahiro Yamasaki received the Dr. degree in Solid State Physics from Osaka University, Japan in 1989.
He joined Fujitsu Laboratories Ltd., Atsugi in 1989, where he has been engaged in first principles molecular dynamics simulations for semiconductor surfaces. From November 1993 until March 1997, he participated in the ATP (Angstrom Technology Partnership).

Kiyoyuki Terakura received the B.E. degree in Physical Engineering from the University of Tokyo, Japan in 1964 and the Dr. degree in Science from Osaka University in 1971. Until 1993, he was working at the Institute for Solid State Physics at the University of Tokyo.
Since 1994, he has been a principal researcher at the National Institute for Advanced Interdisciplinary Research and the Deputy Project Leader of the Joint Research Center for Atom Technology.

Koichi Sato received the B.E. degree in Business Management Engineering from the Science University of Tokyo, Japan in 1982. He joined Toshiba Corporation, Tokyo, Japan in 1982.
From April 1994 until March 1997, he participated in the ATP (Angstrom Technology Partnership), where he has been engaged in management of computer systems, including vector parallel processors (VPPs).