Progress In Electromagnetics Research B, Vol. 27, 327–348, 2011

ACCELERATION OF VORTEX METHODS CALCULATION USING FMM AND MDGRAPE-3

T. K. Sheel

Institute for Modeling and Simulations
University of Rostock, Rostock 18059, Germany

Abstract—The present study discusses some numerical techniques on the simultaneous use of the Fast Multipole Method (FMM) and special-purpose computer (MDGRAPE-3) to make the impractically expensive calculation feasible without the loss of numerical accuracy. In the present calculations, the impingement of two identical inclined vortex rings has been studied, and the computation time has been reduced by a factor of 1000 at $N = 1.18 \times 10^6$ where N is the number of vortex elements. The direct and MDGRAPE-3 calculations both have a scaling of $O(N^2)$, and the use of the FMM brings them both down to $O(N)$. The global kinetic energy, enstrophy and energy spectra have been investigated to address the numerical accuracy and have good agreement with other similar works.

1. INTRODUCTION

The vortex methods have been developed and applied to analysis of complicated, unsteady and vortical flows related with problems in a wide range of industries, because they consist of simple algorithm based on physics of flow. Leonard [1] summarized the basic algorithm and example of its applications. The main difficulties with vortex methods as originally formulated are those of the cost and accuracy of the evaluation of the velocity field induced by N vortices. The cost of the evolution of velocity is $O(N^2)$, and it increases according to the number of vortices N. The vortex method calculation is expensive, particularly in three dimensions where a large number of elements are computed simultaneously, and calculation load becomes impractically expensive.

There are two ways to reduce calculation cost to simulate those complex fluid flows. One is to use fast algorithms such as the

Received 18 September 2010, Accepted 17 November 2010, Scheduled 23 December 2010
Corresponding author: Tarun Kumar Sheel (tarun.sheel@uni-rostock.de).

tree code developed by Barnes and Hut [2] or the fast multipole method (FMM) by Greengard and Rokhlin [3]. The tree code is an $O(N \log N)$ algorithm based on a hierarchical oct-tree representation of space in three dimensions. In the FMM, the long-range forces are approximated by multipole expansion truncated at a certain degree, while the contributions from particles within nearby regions are calculated directly in a usual manner without approximation. In particular, the computation of high-order term is very expensive.

The other way is to execute the flow simulation with special purpose hardware such as MDGRAPE-3 (developed by Narumi et al. [4]). One of the GRAPE (Gravity Pipe, developed by Sugimoto et al. [5]) series machines is a special-purpose computer exclusively designed for molecular dynamics simulations. Its performance is $8 \sim 30$ times higher than MDGRAPE-2 (predecessor of MDGRAPE-3) and $10 \sim 1000$ times higher than general-purpose (defined as 'host' hereafter) computers of the same cost.

The long computation time due to the above-mentioned $O(N^2)$ problem has been reduced 50 times when we apply a special-purpose computer, MDGRPAE-2 shown in [6]. Although it can speed up velocity calculation significantly, the calculation cost is still proportional with N^2 for direct summation algorithm. Implementation of fast algorithms on MDGRAPE-3 therefore is demanded for large scale simulations. The first implementation of fast algorithm (treecode) on GRAPE presented in [7] is 30–50 times faster than the treecode without GRAPE. The second implementation of another fast algorithm (FMM) on MDGRAPE-2 presented in [8] speeds up significantly compared to treecode and direct summation algorithm at high accuracy.

The purpose of the present paper is, therefore, to implement the fast multipole method (FMM) whose time complexity is $O(N)$ on MDGRAPE-3 for the calculation of impingement of two identical inclined vortex rings. Since MDGRAPE-3 can only operate with point-charge or point-mass interaction, we modify the original FMM so that FMM can be run on MDGRAPE-3. This implementation combines advantages of fast algorithm and fast hardware thus delivers good performance for velocity calculation with high accuracy.

This paper presents the implementation, numerical results of accuracy and gives performance comparison with implementation of FMM and direct summation on MDGRAPE-3. The simultaneous use of the above two methods is possible since the MDGRAPE-3 still has a scaling of $O\left(N^2\right)$, and the FMM can be used to reduce it. With the help of these acceleration techniques, the present study compares the time variation of kinetic energy, enstrophy and energy spectra with

other similar works for the same flow field.

2. NUMERICAL METHODS

2.1. Vortex Methods

Vortex methods are part of a wider class of methods: the Lagrangian methods are used to simulate unsteady, convection-dominated, problems. Those are expressed by transport equations written in conservative form, often with a diffusion term, and eventually with a source/depletion term [9]. Here we will briefly discuss about the formulation of three-dimensional vortex element method as follows.

2.1.1. Formulation of 3D Vortex Element Method (VEM)

Vortex element methods have been growing in popularity in last three decades. As their name indicates, they are based on the discretization of vorticity—a quantity that has a compact support in many physical problems—thereby making this approach interesting [10].

The three-dimensional incompressible flow of a viscous fluid has been studied here. The evolution equation for vorticity is

$$\frac{\mathrm{D}\boldsymbol{\omega}_i}{\mathrm{D}t} = (\boldsymbol{\omega}_i \cdot \nabla)\,\mathbf{u} + \nu\nabla^{\mathbf{2}}\boldsymbol{\omega}_{\mathbf{i}} \tag{1}$$

where $\boldsymbol{\omega_i}$ is the vorticity defined as $\boldsymbol{\omega_i} = \nabla \times \mathbf{u}$; $\mathbf{u}$ is the velocity of the vortex element; $(\boldsymbol{\omega} \cdot \nabla)\mathbf{u}$ is called the stretching term and represents the rate of change of vorticity by deformation of vortex lines; ν is the kinematic viscosity; and the term $\nu\nabla^2\boldsymbol{\omega_i}$ represents the change of vorticity by viscous diffusion. The velocity field in a three-dimensional problem is,

$$\mathbf{u}(\mathbf{x}) = -\frac{1}{4\pi}\int \frac{(\mathbf{x}-\mathbf{x}')\times\boldsymbol{\omega}(\mathbf{x}')}{|\mathbf{x}-\mathbf{x}'|^3}\,dV(\mathbf{x}') \tag{2}$$

where $\mathbf{x}$ and $\mathbf{x}'$ are the positions of vortex elements, and dV is the volume of the element. Using the Winckelmans [11] model as a cut-off function:

$$\zeta = \frac{1}{(2\pi\sigma^2)^{d/2}}\exp\left(-\frac{|\mathbf{x}_j-\mathbf{x}_i|^2}{2\sigma^2}\right). \tag{3}$$

Then the Biot-Savart law is formulated as follows

$$\mathbf{u}_i = -\frac{1}{4\pi}\sum_{j=1}^{N}\frac{\mathbf{r}_{ij}^2 + (5/2)\sigma_j^2}{\left(\mathbf{r}_{ij}^2+\sigma_j^2\right)^{5/2}}\mathbf{r}_{ij}\times\gamma_j \tag{4}$$

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, σ_j and $\boldsymbol{\gamma}_j$ are the distances of the position vector, core radius and strength of element. The subscript i stands for the target elements, while j stands for the source elements. The stretching term of Eq. (1) can be discretized as follows:

$$\frac{d\boldsymbol{\omega}_i}{dt} = (\boldsymbol{\omega}_i \cdot \nabla)\,\mathbf{u} \tag{5}$$

If we put vortex strength $\gamma_i = \boldsymbol{\omega}_i d^3\mathbf{x_i}$ in Eq. (5), then it becomes

$$\frac{d\gamma_i}{dt} = (\gamma_i \cdot \nabla)\,u_i \tag{6}$$

Hence, the vortex strength of an individual element is expressed by Eq. (4) in a discretized formulation as

$$\begin{aligned}\frac{d\gamma_i}{dt} \quad = \quad & \frac{1}{4\pi}\sum_{j=1}\left\{-\frac{|\mathbf{r}_{ij}|^2 + (5/2)\sigma_j^2}{\left(|\mathbf{r}_{ij}|^2 + \sigma_j^2\right)^{5/2}}\gamma_i \times \gamma_j\right. \\ & \left. +3\frac{|\mathbf{r}_{ij}|^2 + (7/2)\sigma_j^2}{\left(|\mathbf{r}_{ij}|^2 + \sigma_j^2\right)^{7/2}}\left(\gamma_i \cdot \mathbf{r}_{ij}\right)\left(\mathbf{r}_{ij} \times \gamma_j\right)\right\}\end{aligned} \tag{7}$$

where all notations carry the same meaning as in Eq. (4). For detailed mathematical formulations see in [6, 12].

2.2. Fast Multipole Method

Fast multipole method (FMM) was first presented in [3] for two-dimensional case and was extended to three dimension. The implementation detail for three-dimensional case is given in [13–15]. Here we describe in brief the non-adaptive version of FMM for three-dimensional case of vortex method calculations as follows.

The Biot-Savart Eq. (4)

$$\mathbf{u}_i = \sum_j \boldsymbol{\gamma}_j \zeta \times \nabla G, \tag{8}$$

and the stretching term (6)

$$\frac{D\boldsymbol{\gamma}_i}{Dt} = \boldsymbol{\gamma}_i \cdot \nabla \mathbf{u}_i = \boldsymbol{\gamma}_i \cdot \nabla \sum_j \boldsymbol{\gamma}_j \zeta \times \nabla G = \sum_j \boldsymbol{\gamma}_j \zeta \times \nabla\nabla G \cdot \boldsymbol{\gamma}_i, \tag{9}$$

are calculated using the FMM in order to reduce the complexity from $O(N^2)$ to $O(N)$. G is the Green's function of the Laplace equation, which is $1/4\pi r$ in 3-D. ζ is the cutoff function, which we define in (3).

For the FMM equations, we will adopt the conventions used in [16]. By doing so, the Green's function can be expressed by the multipole expansion

$$\sum_j G = \frac{1}{4\pi} \sum_{n=0}^{p} \sum_{m=-n}^{n} \underbrace{r_i^{-n-1} Y_n^m(\theta_i, \phi_i)}_{S} \underbrace{\left\{ \sum_{j=1}^{N} \rho_j^n Y_n^{-m}(\alpha_j, \beta_j) \right\}}_{M}, \quad (10)$$

and also the local expansion

$$\sum_j G = \frac{1}{4\pi} \sum_{n=0}^{p} \sum_{m=-n}^{n} \underbrace{r_i^n Y_n^m(\theta_i, \phi_i)}_{R} \underbrace{\left\{ \sum_{j=1}^{N} \rho_j^{-n-1} Y_n^{-m}(\alpha_j, \beta_j) \right\}}_{L}. \quad (11)$$

We define the operators S, M, R, L to simplify the equations in the following steps. Using these operators, (8) can be written as

$$\mathbf{u}_i = \frac{1}{4\pi} \sum_{n=0}^{p} \sum_{m=-n}^{n} \boldsymbol{\gamma}_j M \times \nabla S \quad (12)$$

$$\mathbf{u}_i = \frac{1}{4\pi} \sum_{n=0}^{p} \sum_{m=-n}^{n} \boldsymbol{\gamma}_j L \times \nabla R. \quad (13)$$

Similarly, (9) can be written as

$$\frac{D\boldsymbol{\gamma}_i}{Dt} = \frac{1}{4\pi} \sum_{n=0}^{p} \sum_{m=-n}^{n} \boldsymbol{\gamma}_j \nabla M \times (\boldsymbol{\gamma}_i \cdot \nabla S) \quad (14)$$

$$\frac{D\boldsymbol{\gamma}_i}{Dt} = \frac{1}{4\pi} \sum_{n=0}^{p} \sum_{m=-n}^{n} \boldsymbol{\gamma}_j \nabla L \times (\boldsymbol{\gamma}_i \cdot \nabla R). \quad (15)$$

The cutoff function does not appear in these equations since they are used to calculate the effect of the far field, for which it would have negligible effect. A schematic of the flow of calculation is shown in Fig. 1. For example, we want to calculate the induced velocity of the particles in the dark grey box in the first figure. The flow of the calculation is as follows.

Step ① Calculate $\boldsymbol{\gamma}_j M$ for all boxes. The summation in the M operator is for all particles in each box.

Step ② Translate the multipole expansion to the center of larger boxes. The translation operators also follow that of [16].

Step ③ Translate $\boldsymbol{\gamma}_j M$ to $\boldsymbol{\gamma}_j L$. Note that this translation is valid only for non-neighboring boxes.

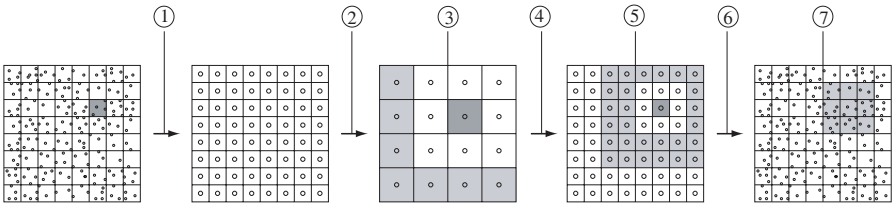


Figure 1. Flow of FMM calculation.

Step ④ Translate the local expansion to the center of smaller boxes.

Step ⑤ Translate $\boldsymbol{\gamma}_j M$ to $\boldsymbol{\gamma}_j L$ for the remaining boxes.

Step ⑥ Calculate the induced velocity using (13).

Step ⑦ Calculate the remaining induced velocity using (8) for all particles in the light grey box in the last figure.

The calculation of the stretching term is done in the same manner by first calculating $\boldsymbol{\gamma}_j \nabla M$, then $\boldsymbol{\gamma}_j \nabla L$ and finally (15) and (9). The schematic in Fig. 1 only shows two levels of box divisions, but the actual calculation requires several more division levels. In such calculations, Steps ②, ④ and ⑤ are calculated more than once. The entire calculation process requires an efficient method for indexing and bookkeeping of the particles and boxes in each level. For the indexing and bookkeeping we adopt the method by Gumerov [17].

The most time consuming parts of the FMM are the multipole to local (M2L) translation (Step ⑤) and the direct calculation (the final step). The balance between these two steps are dependent on the level of box divisions. Dividing the particles into excessively small boxes will result in an enormous amount of multipole to local translations, whereas not dividing them enough would result in a large amount of direct calculation of neighboring particles. These two steps must be balanced by changing the level of box divisions according to the number of particles being calculated. The related discussion can be found in [18–20].

3. HIGH-PERFORMANCE COMPUTING TECHNIQUES

3.1. The MDGRAPE-3 System

MDGRAPE-3 is a successor of MDGRAPE-2 and the third model of MD-GRAPE series. The nominal peak performance of the board for a Coulomb force calculation is 2.16 TFLOPS at 250 MHz for classical molecular dynamics simulations [4, 21]. The basic architecture and

calculation systems are similar to its predecessor, MDGRAPE-2. The brief introduction of this processor is as follows.

The block diagram of an MDGRAPE-3 board can be seen in [4] (Fig. 1). It consists of twelve MDGRAPE-3 chips, and each chip is connected in serial to send/receive the data. Since the memory is embedded in the MDGRAPE-3 chip, the board will be extremely simple. The speed of the communication between the chips will be 1.3 Gbytes/sec, which corresponds to an 80-bit word transfer at 133 MHz. For these connections 1.5 V-CMOS I/O cells will be used. The board has a control FPGA (or ASIC) with a special bus with 1 Gbytes/sec peak speed.

3.1.1. Calculations System

Figure 2 depicts the calculation systems of this special-purpose computer. One small MDGRAPE-3 board (consists of 2 chips) has the peak performance of 330 GFlops. In order to communicate with the host computer, a field-programmable gate array (FPGA, Xilinx XC2VP30) is installed on the board. It also controls the chips, thermal sensors, and so on. The board is connected to the host by a 10-Gbps serial communication link with a 4-lane 2.5-Gbps Xilinx RocketIO through an InfiniBand cable. The host computer has an interface card with an FPGA (Xilinx XC2VP7) attached to a PCI-X bus (Fig. 2). For the present calculations, Xeon 5160 (3.0 GHz) duel core processor has been used as a host PC. The calculation and data transfer systems are the same as that of MDGRAPE-2. It can speed up force calculation about 100–1000 times faster than that of general purpose computers of the same cost. The time complexity of force calculation is $O(N^2)$ for direct summation algorithm.

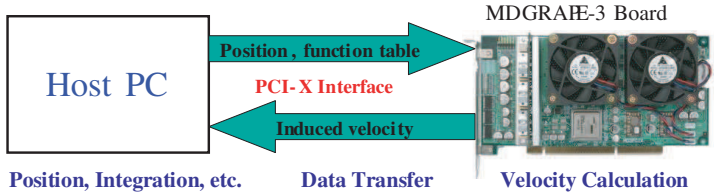


Figure 2. The basic structure of vortex methods calculation inMDGRAPE-3 system.

3.2. FMM on MDGRAPE-3

The MDGRAPE-3 is the third model of a special purpose computer designed for molecular dynamics calculations. The vortex method has been used on the second model MDGRAPE-2 by the present author [6] for calculating the collision of vortex rings. The present study focuses on the simultaneous use of the FMM and MDGRAPE-3 for the same flow field. One major problem in this sense is that the MDGRAPE chips can only handle two types of calculations. The Coulomb potential

$$p_i = \sum_{j=1}^{N} b_j g\left(a|\mathbf{r}_{ij}|^2\right), \tag{16}$$

and Coulomb force

$$f_i = \sum_{j=1}^{N} b_j g\left(a|\mathbf{r}_{ij}|^2\right) \mathbf{r}_{ij}. \tag{17}$$

$g()$ is an arbitrary function, which must be defined prior to the calculation. a and b_j are constants, which can be used for scaling. The direct form of the Biot-Savart Eq. (8) and the stretching term (9) can be calculated by using a combination of (16) and (17), but the multipole and local expansions and their translations are impossible to calculate. Therefore, the MDGRAPE-3 can only be used for the final step of the FMM where it calculates the direct interaction of particles. This in turn will prevent the optimum level of box divisions from growing, and this method will no longer have a complexity of $O(N)$.

The inefficiency of the above method resides in the fact that only one of the two hot spots of the FMM is calculated on the MDGRAPE-3. It is possible to calculate both hot spots on the FMM if we can convert the multipole to local translation into a direct calculation. This requires the use of two independent methods, the Poisson integral method [22] and pseudo-particle method [23]. Instead of calculating the multipole and local expansions at the center of the boxes, these methods calculate the physical properties of interest at quadrature points placed on a spherical shell surrounding the boxes. In contrast to the original FMM, which uses 5 different equations for the expansions and translations, these methods use only 2, one for the multipole translation

$$Q_i = \sum_{j=1}^{N} q_i \sum_{n=0}^{p} \frac{2n+1}{K} \left(\frac{\rho_j}{r_s}\right)^n P_n\left(\cos\gamma_{ij}\right) \tag{18}$$

and another for local translations

$$Q_i = \sum_{j=1}^{N} q_i \sum_{n=0}^{p} \frac{2n+1}{K} \left(\frac{r_s}{\rho_j}\right)^{n+1} P_n\left(\cos\gamma_{ij}\right). \tag{19}$$

The notations still follow that of [16], but additional variables have been introduced. Q and q are the physical properties of interest, which are the potential for Anderson's method and circulation for Makino's method. Q represents the physical property after the translation and q represents the one before. K is the number of quadrature points on the sphere surrounding the box, so the index i runs from 1 to K, and r_s is the radius of this sphere. γ_{ij} is the angle between the position vector of source and target particles. Given that $\mathbf{x}_i = (r_i, \theta_i, \phi_i)$ and $\mathbf{x}_j = (\rho_j, \alpha_j, \beta_j)$, γ_{ij} can be written as

$$\cos\gamma_{ij} = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{r_i \rho_j} \tag{20}$$

Next, we will give a brief explanation of how these two methods are actually used, by considering an example analogous to the one shown in Fig. 1. We will assume that we are calculating the Biot-Savart equation. The flow of calculation is shown in Fig. 3. In this procedure the potential equation

$$\mathbf{\Phi}_i = \sum_{j=1}^{N} \frac{\boldsymbol{\gamma}_j}{4\pi r_{ij}} \tag{21}$$

is also calculated.

Step ① (21) is calculated for the quadrature points on a large sphere, having a radius twice as that of the circumscribing sphere. Then we solve a system of equations to calculate the circulation of the quadrature points on the circumscribing sphere.

Step ② Makino's method (18) is used to translate the circulation onto the larger spheres.

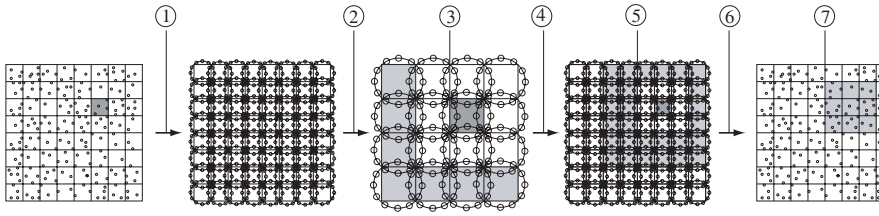


Figure 3. Flow of FMM calculation without multipoles.

Step ③ (21) is calculated for the quadrature points on the non-neighboring spheres. In the corresponding figure only the adjacent boxes (3×3) are defined as neighbors, which is different from the actual case. Since the interaction is calculated between the quadrature points on the circumscribing sphere instead of the multipole moments at the center of the box, the quadrature points become too close for a far field approximation. In our method we expand the definition of neighbors to a larger region (5×5) to retain accuracy. Consequently, the coarsest level in this calculation should be level 3 instead of 2.

Step ④ Anderson's method (19) is used to translate the potential onto the smaller spheres.

Step ⑤ (21) is calculated for the quadrature points on the remaining non-neighboring spheres. The neighbor region is 5×5, so the number of source boxes can reach 875 per target. This increase is quite large and will slow down the method considerably as we will show later.

Step ⑥ Solve a system of equations to determine the circulation of the quadrature points on a large sphere, having a radius twice as that of the circumscribing sphere. Then, calculate (8) to obtain the velocity of all particles in the corresponding box.

Step ⑦ Calculate the remaining induced velocity using (8) for all particles in the light grey box in the last figure.

Now, the two most time consuming Steps ⑤ and ⑦ can both be calculated on the MDGRAPE-3. In this case the complexity of the calculation should remain $O(N)$. We also constructed a pseudo-particle tree code by skipping Steps ③, ④, and ⑤ and directly calculating the local expansion at each level.

3.3. Hot-spot of FMM Calculation

Here we focus on the hot-spot of FMM calculation and the possibility to use MDGRAPE-3. Fig. 4 represents the hot spots of FMM calculation. If we want to calculate for light gray box, far particles are solved by FMM, and the neighbor particles are solved directly. First step is to calculate Multipole to Multipole (called M2M) translation that is the summation for all particles in each box and translate the multipole expansion to the center of larger boxes. Then it performs the multipole to local (M2L) translation. Here red is source particle, and blue is target. Note that M2L cannot be preformed for neighbors. The next step is to translate the local expansion to the center of smaller boxes. Then again preform M2L calculation for the remaining boxes.

Calculate the induced velocity up to this step. Many sources act on one target in this step, and this is one of the hot-spots of the FMM calculation. Finally, calculate the remaining induced velocity by direct calculation for all particles in the light gray box. If the box is too coarse this could also be the hot-spot of the FMM calculation.

The most time consuming parts of the FMM are the multipole to local (M2L) translation and the direct calculation. The balance between these two steps is dependent on the level of box divisions. Dividing the particles into excessively small boxes will result in an enormous amount of multipole to local translations, whereas not dividing them enough would result in a large amount of direct calculation of neighboring particles. These two steps must be balanced by changing the level of box divisions according to the number of particles being calculated.

The last plot (most left of below) of Fig. 4 shows the CPU time for different steps of FMM calculations. It can be easily observed that the Direct and M2L calculations consumed most of the time. The CPU time of the rest steps is negligible. It is necessary to balance the cost of direct and M2L calculations.

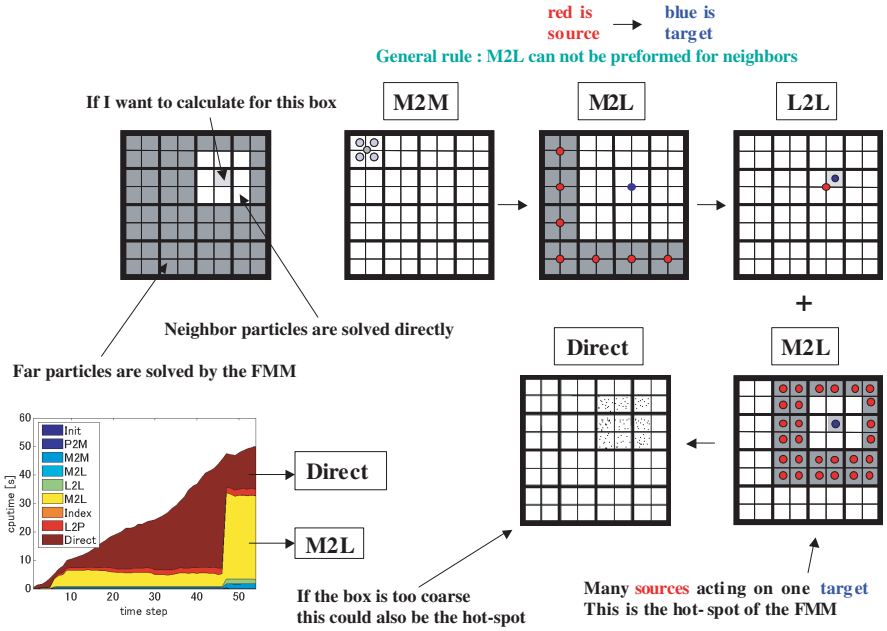


Figure 4. Hot-spot of FMM calculation.

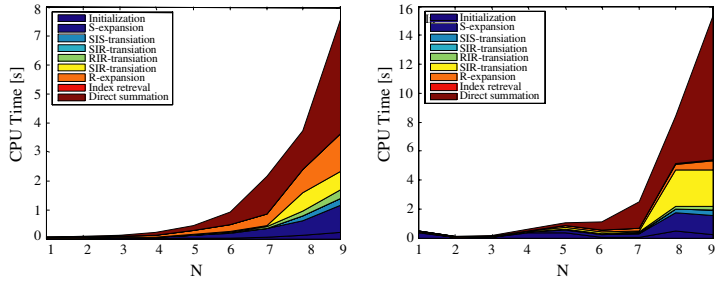


Figure 5. Elapsed time on MDGRAPE-3 (Biot-Savart).

The elapsed time has been investigated for every step of FMM calculation with and without the use of MDGRAPE-3. This analysis will lead to estimating and optimizing the calculation cost and use of both techniques to implement in actual vortex method calculations.

Figure 5 shows the elapsed time to implement the FMM and the FMM on MDGRAPE-3 for Biot-Savart calculation which calculates induced velocity from vorticity transport equation, Eq. (1). Elapsed time of different calculations are represented in different colours. It can be easily seen that the brown colour consumed half of the total time in both cases which stands for direct calculation. This means that it is required to reduce direct calculation time to get the maximum efficiency of these methods. In Fig. 5(b), multipole to local translation (M2L) consumed second most time, contrary to FMM on MDGRAPE-3. This may cause the limitations of MDGRAPE-3 hardware specifications.

3.4. Performance

The calculation cost and accuracy are an important issue for any numerical simulation. In this calculation these two factors have been investigated carefully. The calculation has been accelerated retained the accuracy at an acceptable level. The CPU time has been compared with different acceleration techniques at one time step by changing the number of particles.

The L^2 norm error is defined as the difference in the induced velocity of the same particles between the host and MDGRAPE for the same time step as follows.

$$L^2(norm\ error) = \frac{\sum \left((u_{host} - u_{md})^2 + (v_{host} - v_{md})^2 + (w_{host} - w_{md})^2\right)}{\sum \left(u_{host}^2 + v_{host}^2 + w_{host}^2\right)} \tag{22}$$

where the suffices *md* and *host* are represented with and without the use of MDGRAPE, respectively.

The Biot-Savart and stretching term calculations are performed separately and evaluate the CPU time for different element numbers. The same calculations are done with and without the MDGRAPE-3. The CPU time of the Biot-Savart and Stretching term calculation for one time step is plotted against the number of elements in Fig. 6. The L^2 norm error between the direct calculation and FMM calculations are also shown in Fig. 7.

The CPU time for all methods (when optimized) are plotted in Fig. 6. Xeon 5160, MDG3, FMM, and FMM-MDG3 represent the calculation without FMM or MDGRAPE-3, with MDGRAPE-3, with FMM, with FMM and MDGRAPE-3. The direct calculation without MDGRAPE-3 has a high asymptotic constant and an order of $O(N^2)$. All calculations were performed on a dual core Xeon 5160 (3.0 GHz) processor. The direct calculation on MDGRAPE-3 has a low asymptotic constant but still has a scaling of $O(N^2)$. On the contrary, the FMM without MDGRAPE-3 has a high asymptotic constant, but

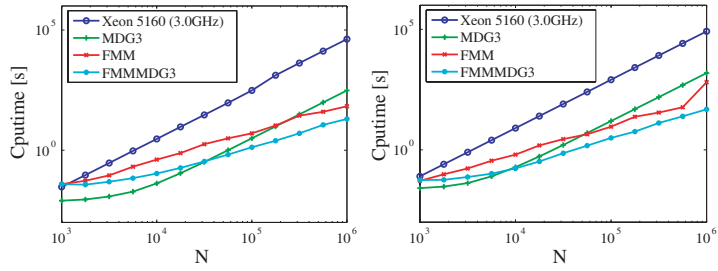


Figure 6. Cpu-time of FMM, MDGRAPE-3, and both.

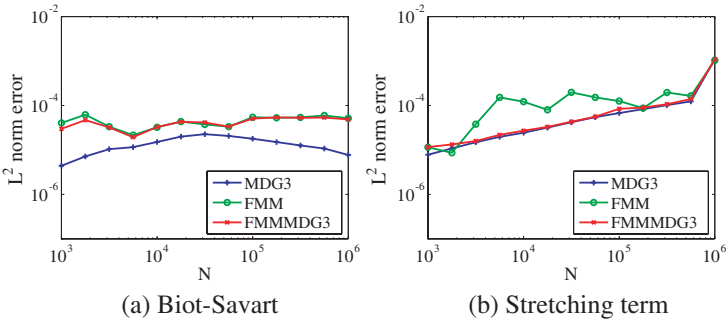


Figure 7. Accuracy of FMM, MDGRAPE-3, and both.

its complexity is $O(N)$ in both cases (Figs. 6(a) and 6(b)). The combination of FMM and MDGRAPE-3 results in a calculation with a low asymptotic constant and $O(N)$ complexity. At $N = 10^6$ the FMM on MDGRAPE-3 is approximately 4 times faster than the FMM.

The cost of original FMM was $O(N \log N)$, but in our case it is close to $O(N)$, which is probably due to the order of multipole moment and MDGRAPE-3. We have set the order of moment $P = 10$ in all calculations. This number has been set after rigorous investigation of the accuracy and calculation time. The "P" has great effect on the accuracy of FMM calculation.

The accuracy as a L^2 norm error achieved by the combination of FMM and MDGRAPE-3 has been plotted for the Biot-Savart and stretching term calculations, without any acceleration, with FMM, MDGRAPE-3, and both FMM and MDGRAPE-3 in Fig. 7. The MDGRAPE-3 has a small error compared to FMM and in combination with both in Fig. 7(a). The error of MDGRAPE-3 is controlled by its system, and using FMM this error slightly increases as in the same order of magnitude compared with FMM. The calculation time takes longer, and the L^2 norm error becomes larger for large N in the case of stretching term calculations (Figs. 6(b) and 7(b)) compared with Biot-Savart calculation. This error may be caused by the vortex strength and the discretization error of stretching term.

The quantitative acceleration ratio for $N = 10^6$ is given in Table 1. For Biot-Savart calculation, the FMM alone accelerates the calculation 462 times, and simultaneous use of the MDGRAPE-3 further accelerates it 4.1 times. From a different perspective, the MDGRAPE-3 can accelerate the calculation 119 times, but the simultaneous use of the FMM allows a 16 fold increase from that. Similarly, the stretching term calculation is 613 times faster when we use the FMM, and another 2.8 times faster when we combine it with the MDGRAPE-3. The MDGRAPE-3 accelerates the stretching term calculation 52 times, and another 33 times if we use the FMM on it.

Table 1. Accelaration ratio at $N = 10^6$.

Biot-Savart		stretching	
direct		direct	
↓ ×462	↓ ×119	↓ ×613	↓ ×52
FMM	MDG3	FMM	MDG3
↓ ×4.1	↓ ×16	↓ ×2.8	↓ ×33
FMM + MDG3		FMM + MDG3	

4. APPLICATIONS

4.1. Calculation Conditions

We considered inclined collisions according to [6] to check and validate our new scheme. Here we assumed that the initial radius of the vortex rings was $R = 1$ while the cross-section radius $r = 0.05$ (see Fig. 8). The Reynolds number based on the ring circulation was $\mathrm{Re}_\Gamma = 400$, and the core radius $\sigma = 0.065$. The rings were inclined at an angle $\theta = 15^\circ$ relative to the z-axis. The total number of elements used for the previous calculation was $N = 6 \times 10^4$, with the number of cross sections in the circumference direction being 502, while 61 elements were distributed in each cross-section. The total number of elements used for the present calculation was $N = 1 \cdot 2 \times 10^6$, with the number of cross sections in the circumference direction being 1495, while 397 elements were distributed in each cross-section. All elements were evenly distributed. This number was 2 to 15 times larger than the previous calculations done by the present author and the other researchers for the same flow field.

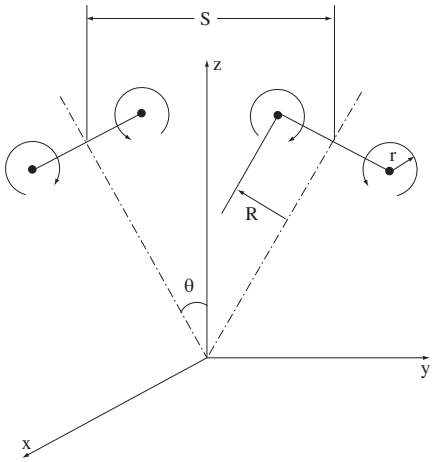


Figure 8. Initial condition for the computation of the collision of two vortex rings. Here R — radius of ring, $\mathbf{r}$ — radius of cross-section, S — distance between two rings, θ — inclined angle.

In this calculation, the viscous diffusion was calculated using the core-spreading method developed by Leonard [1]. For convection of the particles, the second order accurate Adams-Bashforth method was used in the calculation of time advances.

The error was defined as the difference in the induced velocity of

the same particles between the host and MDGRAPE for the same time step as of Eq. (22). Here error was defined as the distance δ instead of L^2.

The kinetic energy E and enstrophy Ω are evaluated from the particle positions and strengths according to [11], which are defined as follows.

$$E = \frac{1}{16\pi} \sum_{i,j} \left[\frac{2\left(\gamma_i \cdot \gamma_j\right)}{\left(\mathbf{r}_{ij}^2 + \sigma_j^2\right)^{1/2}} + \frac{\left(\mathbf{r}_{ij} \cdot \gamma_i\right)\left(\mathbf{r}_{ij} \cdot \gamma_j\right) - \mathbf{r}_{ij}^2\left(\gamma_i \cdot \gamma_j\right)}{\left(\mathbf{r}_{ij}^2 + \sigma_j^2\right)^{3/2}} \right] \quad (23)$$

and

$$\begin{aligned} \Omega \;=\; & \frac{1}{4\pi} \sum_{i,j} \left[\frac{5\sigma_j^4 - \mathbf{r}_{ij}^2\left(\mathbf{r}_{ij}^2 + 3.5\sigma_j^2\right)}{\left(\mathbf{r}_{ij}^2 + \sigma_j^2\right)^{7/2}} \left(\gamma_i \cdot \gamma_j\right) \right. \\ & \left. + 3\frac{\left(\mathbf{r}_{ij}^2\left(\mathbf{r}_{ij}^2 + 4.5\sigma_j^2\right) + 3.5\sigma_j^4 i\right)\sigma_j^2}{\left(\mathbf{r}_{ij}^2 + \sigma_j^2\right)^{9/2}} \left(\mathbf{r}_{ij} \cdot \gamma_i\right)\left(\mathbf{r}_{ij} \cdot \gamma_j\right) \right] \end{aligned} \quad (24)$$

The energy spectra are calculated along the z-axis at selected times.

4.2. Results and Discussion

The main focus of this article is to accelerate the calculation by simultaneous use of FMM and MDGRAPE-3. Before analyzing the new scheme we first confirmed our previous calculations for this model as follows one by one. Fig. 9(a) shows the calculation time against the number of vortex elements with and without the use of MDGRAPE. The legends 'Host2', 'Host3', 'MDG2', and 'MDG3' correspond to the calculations without and with the use of MDGRAPE-2 and MDGRAPE-3, respectively. It is clearly observed that the calculation time is reduced by a factor of 50 and 100 with the use of MDGRAPE-2 and -3, respectively for $N \sim 10^5$. On the other hand, MDGRAPE-3 is 25 times faster than MDGRAPE-2 with the same cost. It is important to produce an optimum function table in order to calculate the cut-off function by considering the computational domain where the vortex elements are distributed. The function $g\left(w\right)$ (Eqs. (16) and (17)) is created prior to calculation and read during calculation. The domain of the function $g\left(w\right)$ is set to $w_{\min} \leq w \leq w_{\max}$ where $\left(w_{\max}/w_{\min}\right) \leq 2^{32}$ according to hardware specifications, for details see in [6, 12]. Fig. 9(b) represents the ranges and scaling errors of function table where the x-axis stands for the range of the function $g(w)$ and the y-axis for induced

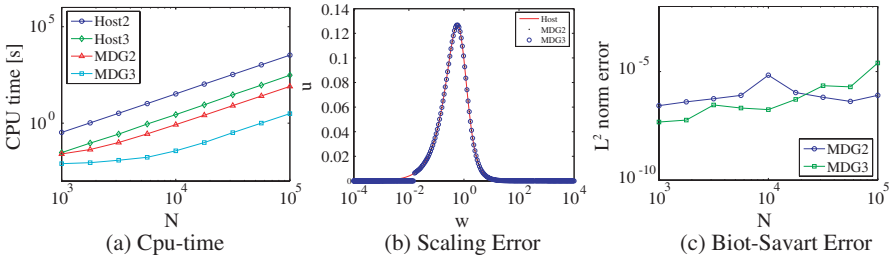
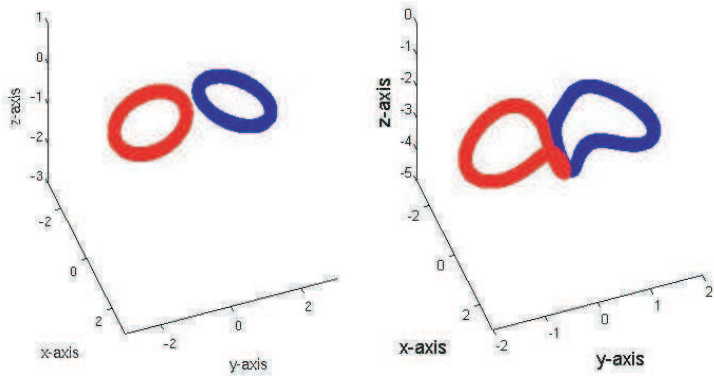


Figure 9. Acceleration and accuracy of MDGRAPE-2 and MDGRAPE-3.

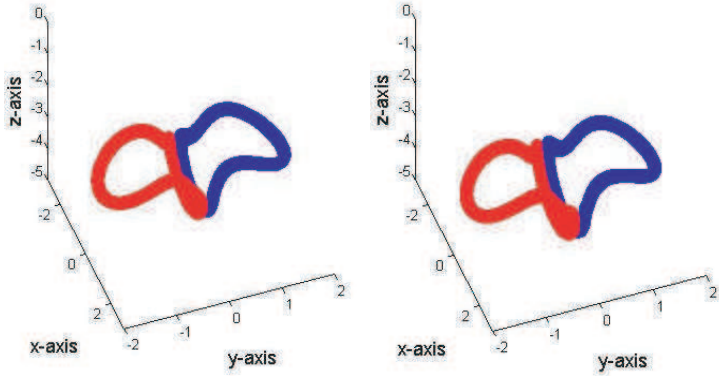
velocity **u**. In our calculation, the table has been produced within the finite range of $2^{-12} \leq w \leq 2^{20}$ which satisfies the computational domain of original calculation to obtain significant accuracy. This figure also presents the error of the induced velocity which affects the convection of vortex elements and generates discrepancies within the calculation without MDGRAPE where they coincided each other. Fig. 9(c) presents the maximum error (22) for Biot-Savart calculation compared between MDGRAPE-2 and MDGRAPE-3. It is shown that both error are in similar order of magnitude. MDGRAPE-3 gives less error for small number of elements while large error for large number compared with MDGRAPE-2. It may be caused by slightly different hardware specifications.

Figure 10 shows the instantaneous flow patterns of vortex elements of two colliding vortex rings in various time stages. The initial setup in colliding ring simulations consists of two identical vortex rings initially inclined at an angle $\theta = 15°$. In Fig. 10(a), the rings are initially placed at a non-dimensional distance of $s = 2.4$ in the z-direction. Each vortex ring is approached by self-induced velocity from this initial stage. At $t\Gamma/R^2 = 5$ in Fig. 10(b), the first impact occurs, and the two vortex rings are stretched and deformed. As time progresses, considerable differences appear in each stage. At $t\Gamma/R^2 = 10$, the arced-shape structure is formed, and the downward stretch is strong (see Fig. 10(d)).

The evolution of the kinetic energy and enstrophy of Host, FMM, MDGRAPE-2, and MDGRAPE-3 compared with Winckelmans work are shown in Fig. 11. In the present calculations the flow is incompressible and unbounded, so there are physically no kinetic energy sources. The kinetic energy can be dissipated by both viscosity and numerical errors. From the comparison between the results obtained with the various time steps, it has been observed that there is no significant difference between host calculation and others. It is



(a) $t\Gamma/R^2 = 1$ (b) $t\Gamma/R^2 = 5$

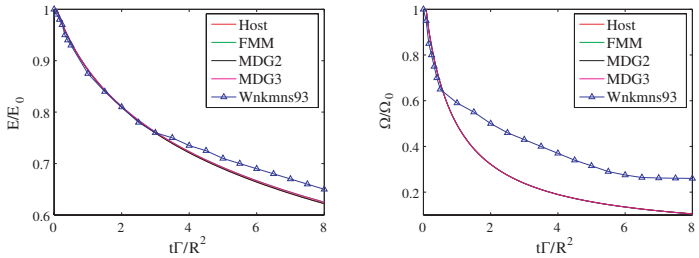


(c) $t\Gamma/R^2 = 8$ (d) $t\Gamma/R^2 = 10$

Figure 10. Instantaneous flow patterns ($N = 1e5$).

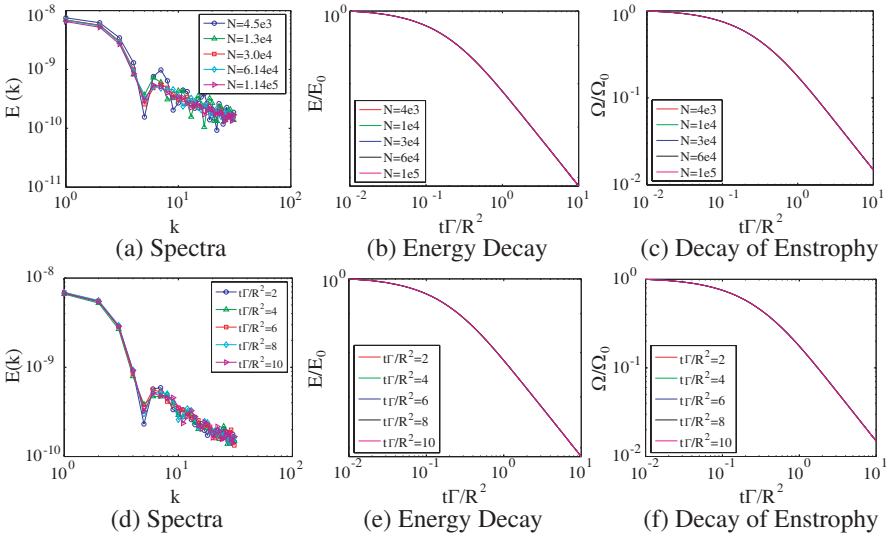
shown in Fig. 11(a) for the time step $\Delta t = 0.08$, and it is easily observed that the agreement with the existing data of [11] is also satisfactory. The difference in the decay of enstrophy (see Fig. 11(b)) is observed between the present computation and that by [11], though this is due to the difference in the treatment of viscous diffusion schemes. A similar tendency has also been observed in the computation of two-dimensional homogeneous isotropic turbulence by [24]. Thus the overall tendency of the present calculation is satisfactory for the assessment of the MDGRAPE-3 and FMM.

Figure 12 presents the evolution of energy spectra and corresponding decay of energy and enstrophy for a different number of particles at the same time step. Figs. 12(a) to (c) present compared results for a different number of elements at $t\Gamma/R^2 = 10$



(a) Kinetic Energy (b) Enstrophy

Figure 11. Time history of kinetic energy & enstrophy.



(a) Spectra (b) Energy Decay (c) Decay of Enstrophy

(d) Spectra (e) Energy Decay (f) Decay of Enstrophy

Figure 12. Energy spectrum and decay of energy & enstrophy.

while Figs. 12(d) to (f) represent different time stages for $N = 1e5$. Figs. 12(a) and (d) show the one-dimensional longitudinal power energy spectra by the core spreading method. The spectra are calculated from the velocity distribution along z-axis. When the energy spectra from $N = 4e3$ to $N = 1e5$ and at $t\Gamma/R^2 = 2.0$ is compared to the one at $t\Gamma/R^2 = 10.0$, the spectra are extended to higher wave number region as elements increase and time progress. The effect of a large number of elements appears in high wave number regions in Fig. 12(d), and the spectrum at $t\Gamma/R^2 = 10.0$ comes up to $k = 32$ and then declines. It is well known that the energy spectra based on the Kolmogorov universal scale at high wave regions have the particular tendency, which is independent of the flow characteristics. According to the results in Figs. 12(a) and (d), the energy spectra

decrease with wave numbers, which means that the energy cascade mechanism is reasonably simulated. In the present calculation, the energy spectra diverge with increased number of particles and progress of time. This is the common behavior of core spreading method without splitting or merging for large number of particles as noted by Cottet [25] and Fukuda [26]. Our results have similar behaviour up to some extent to the work of [26] (not shown here) even the initial conditions are different. The decay of kinetic energy and enstrophy in both cases have the same rate and similar tendency to [24] for two-dimensional homogeneous isotropic turbulence. In both cases, the results are normalized by the initial result of Kinetic energy and enstrophy. Further improvement is considered by using splitting and merging of core spreading method at the first place in the subsequent work.

5. CONCLUSIONS

Vortex method calculations were accelerated by the simultaneous use of the fast multipole method and MDGRAPE-3. Out of the 4 fast algorithms we compared, the rotation based FMM with direct summation on the MDGRAPE-3 was the fastest. The FMM on MDGRAPE-3 is 16 times faster than the MDGRAPE-3 itself, and 4.1 times faster than the FMM on a Intel Xeon (3 GHz) for the Biot-Savart calculation of $N = 10^6$ elements. The time evolution of kinetic energy and enstrophy have been calculated and retained the numerical accuracy at certain level. The analyzed energy spectra have been successfully simulated by using this acceleration techniques.

ACKNOWLEDGMENT

The author thanks his Ph.D. supervisor, Prof. S. Obi, Keio University, Japan for valuable guidance during this work. He also thanks Prof. Kenji Yasuoka, Keio University, Japan for his support from computational tools especially MDGRAPEs machine. The author is very much grateful for financial supports from Yoshida Scholarship Foundation and Amano Scholarship Foundation in Japan during his Ph.D. research.

REFERENCES

1. Leonard, A., "Vortex methods for flow simulations," *J. Comput. Phys.*, Vol. 37, 289–335, 1980.
2. Barnes, J. E. and P. Hut, "A hierarchical $O(N \log N)$ force calculation algorithm," *Nature*, Vol. 324, 446–449, 1986.

3. Greengard, L. and V. Rokhlin, "A fast algorithm for particle simulations," *J. Comput. Phys.*, Vol. 73, 325–348, 1987.
4. Narumi, T., Y. Ohno, N. Okimoto, T. Koishi, A. Suenaga, N. Futatsugi, R. Yanai, R. Himeno, S. Fujikawa, M. Ikei, and M. Taiji, "A 55 TFLOPS simulation of amyloid-forming peptides from yeast prion sup35 with the specialpurpose computer system MDGRAPE-3," *Proceedings of the SC06 (High Performance Computing, Networking, Storage and Analysis)*, CDROM, Tampa, USA, 2006.
5. Sugimoto, D., Y. Chikada, J. Makino, T. Ito, T. Ebisuzaki, and M. Umemura, "A special-purpose computer for gravitational many-body problems," *Nature*, Vol. 345, 33–35, 1990.
6. Sheel, T. K., K. Yasuoka, and S. Obi, "Fast vortex method calculation using a special-purpose computer," *Computers and Fluids*, Vol. 36, 1319–26, 2007.
7. Makino, J., "Treecode with a special-purpose processor," *Pub. of the Astronomical Society of Japan*, Vol. 43, 621–638, 1991.
8. Chau, N. H., A. Kawai, and T. Ebisuzaki, "Implementation of fast multipole algorithm on special-purpose computer MDGRAPE-2," *Proc. of the 6th World Multiconference on Systematics, Cybernetics and Informatics SCI 2002'*, Vol. XVI(2002), 477–481, USA, 2002.
9. Shankar, S., "A new mesh-free vortex method," Ph.D. Thesis, The Florida State University, 1996.
10. Chatelain, P., "Contributions to the three-dimensional vortex element method and spinning bluff body flows," Ph.D. Thesis, California Institute of Technology, 2005.
11. Winckelmans, G. S. and A. Leonard, "Contributions to vortex particle methods for the computation of three-dimensional incompressible unsteady flows," *J. Comput. Phys.*, Vol. 109, 247–273, 1993.
12. Sheel, T. K., R. Yokota, K. Yasuoka, and S. Obi, "The study of colliding vortex rings using a special-purpose computer and FMM," *Transactions of the Japan Society for Computational Engineering and Science*, Vol. 2008, 20080003, 2008.
13. Greengard, L. and V. Rokhlin, *Rapid Evaluation of Potential Fields in Three Dimensions*, in Vortex Methods, Edited by C. Anderson and C. Greengard, Number 1360 in Lecture Notes in Mathematics, 121–141, Springer-Verlag, Berlin, 1988.
14. Sanjay, V. and W. C. Chew, "Analysis and performance of a distributed memory multilevel fast multipole algorithm," *IEEE Trans. Antennas Propag.*, Vol. 53, 2719–2727, 2005.

15. Chew, W. C., J. M. Jin, and M. Eric, *Fast and Efficient Algorithms in Computational Electromagnetics*, Artech House Publishers, 2001.
16. Cheng, H., L. Greengard, and V. Rokhlin, "A fast adaptive multipole algorithm in three dimensions," *J. Comp. Phys.*, Vol. 155, 468–498, 1999.
17. Gumerov, N. A. and R. Duraiswami, *Fast Multipole Methods for the Helmholtz Equation in Three Dimensions*, Elsevier, 2004.
18. Xu, K., D. Z. Ding, Z. H. Fan, and R. S. Chen, "Multilevel fast multipole algorithm enhanced by GPU parallel technique for electromagnetic scattering problems," *Microwave and Optical Technology Letters*, Vol. 53, 502–507, 2010.
19. Ravnik, J., S. Leopold, and Z. Zoran, "Fast single domainsubdomain BEM algorithm for 3D incompressible fluid flow and heat transfer," *IJNME*, Vol. 77, 1627–1645, 2009.
20. Rui, P.-L., R.-S. Chen, Z.-W. Liu, and Y.-N. Gan, "Schwarz-Krylov subspace method for MLFMM analysis of electromagnetic wave scattering problems," *Progress In Electromagnetics Research*, Vol. 82, 51–63, 2008.
21. Taiji, M., T. Narumi, Y. Ohno, N. Futatsugi, A. Suenaga, N. Takada, and A. Konagaya, "Protein explorer: A petaflops special-purpose computer system for molecular dynamics simulations," *Proc. Supercomputing*, in CD-ROM, USA, 2003.
22. Anderson, C. R., "An implementation of the fast multipole method without multipoles," *SIAM J. Sci. Stat. Comput.*, Vol. 13, 923–947, 1992.
23. Makino, J., "Yet another fast multipole method without multipoles-pseudo-particle multipole method," *J. Comput. Phys.*, Vol. 151, 910–920, 1999.
24. Totsuka, Y. and S. Obi, "A validation of viscous dissipation models for fast vortex methods in simulations of decaying turbulence," *Journal of Fluid Science and Technology*, Vol. 2, No. 1, 248–257, 2007.
25. Cottet, G.-H., B. Michaux, S. Ossia, and G. VanderLinden, "A comparison of spectral and vortex methods in three-dimensional incompressible flows", *J. Comp. Phys.*, Vol. 175, 702–712, 2002.
26. Fukuda, K. and K. Kamemoto, "Application of a redistribution model incorporated in a vortex method to turbulent flow analysis," *The 3rd International Conference on Vortex Flows and Vortex Methods*, 131–136, Japan, 2005.