CADMAS-SURF 3D-MG

Chapter 1 **Overview**

(1) Policy

The three-dimensional code (CADMAS - SURF / 3D) developed on the basis of CADMAS - SURF 1) for 2 dimensions was parallelized by the following policy (CADMAS-SURF/3D-MG).

(a) Program Language

Fortran 90 is used for the development language of the analysis section to "take dynamic array" and "make the number of characters of the name longer than 6 characters". However, normal coding conforms to FORTRAN 77.

(b) Parallelization

Parallelize using the Single Program Multiple Data (SPMD) model where the same program runs on multiple nodes For data handling, we adopt a region segmentation method suitable for large-scale calculation.

(c) Parallel computing machine to target

It targets PC clusters with multiple PCs connected to the network. Therefore, the type of parallel computer is assumed to be a distributed memory type. The message exchange library shall be MPI (Message Passing Interfase).

(2) System configuration

The parallel version (CADMAS-SURF/3D-MG) also operates with one processor, in which case it is made to behave like the standalone version (CADMAS - SURF / 3D). Therefore, the system configuration of CADMAS-SURF/3D-MG is divided into two cases: single processor case (Figure 0-1-1) and multiple processor case (Figure 0-1-2).

In the three-dimensional calculation, since the data becomes large scale, we added a graphic file that outputs results only for the specified area, and decided to use it for drawing. In addition, detailed files that output the results of all areas are used only for restarting.

(3) Function

The function list is shown in Table 0-1-1. In the CADMAS - SURF, we decided to delete wave - forming functions of low frequency, QUICK scheme, piston type and flap type.

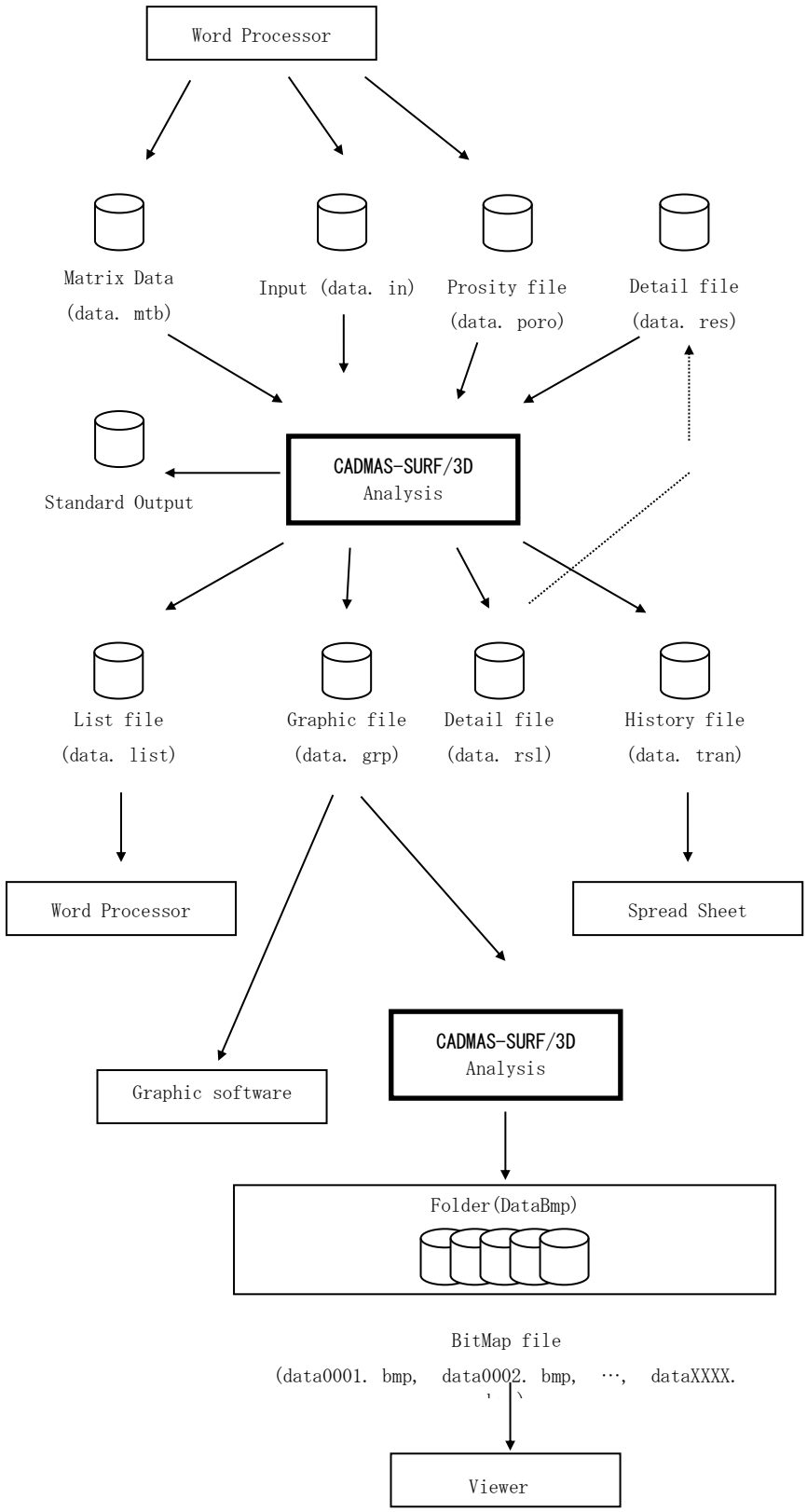


Figure 0-1-1 System configuration for single processor

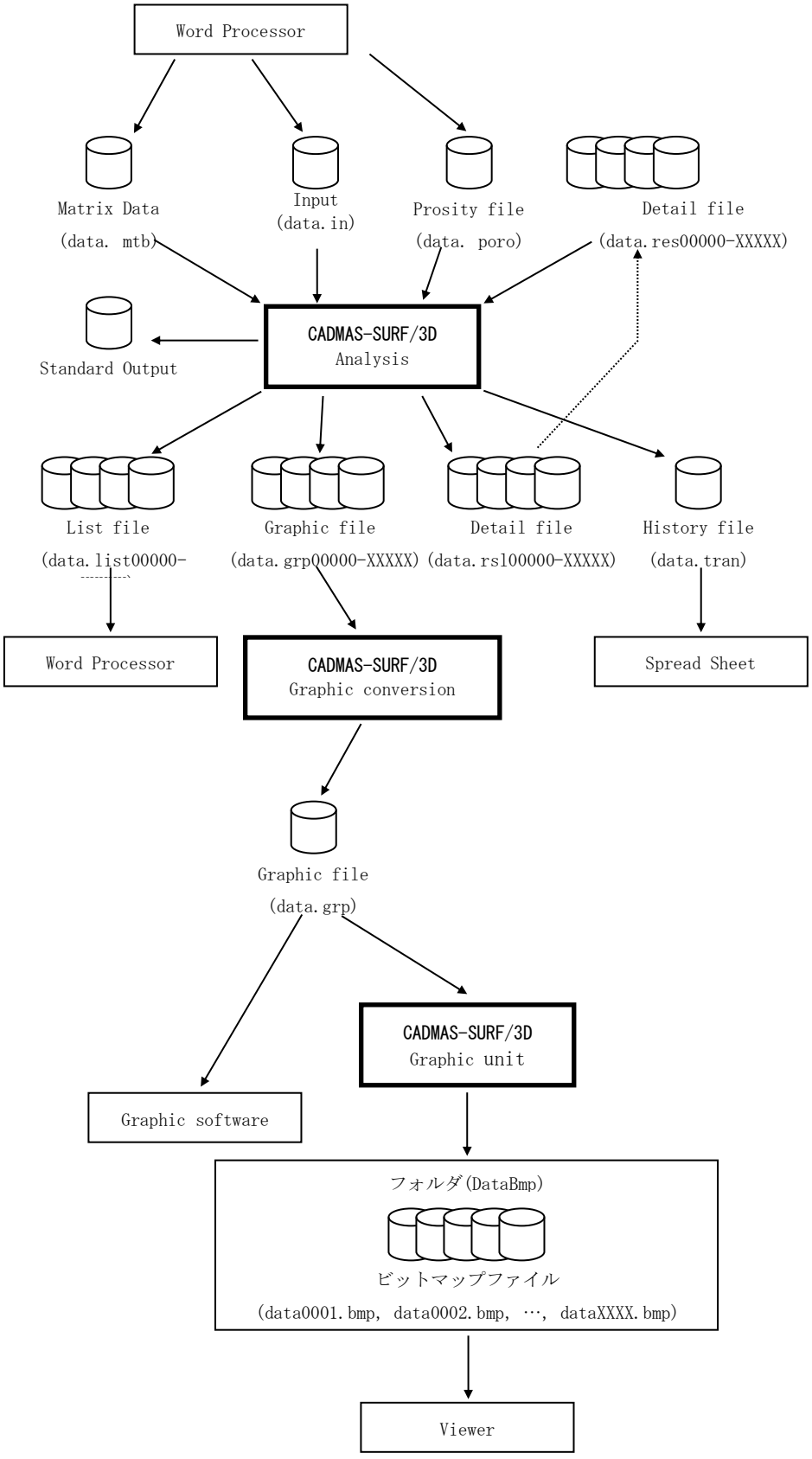


Figure 0-1-2　System configuration for multiple processors

Table 0-1-1　Function list

	Item	Description
Physical model	Target to analyze	Complex flow with free surface as multivalent function
	Basic equations	Expanded expression based on porous model with Navier-Stokes equations of three-dimensional incompressible viscous fluid and continuous equation
	Coordinate system	Cartesian coordinates
	Free surface analysis model	Volume of fluid method（VOF method）
	Turbulence model	High Reynolds number $k-\varepsilon$ model
	Wave model	Waveform boundary Wave source (The following wave function can be applied to each wave model)
	Waveform function	Stokes wave fifth approximation solution (constant traveling wave) Knoid wave third order approximation solution (constant traveling wave) Numerical solution by flow function method B (constant traveling wave) Matrix data (arbitrary waveform)
	Nonreflective model	Radiation boundary of Sommerfeld Energy damping zone
	Scalar amount of advection diffusion	Energy equation Multicomponent concentration transport equations
	General boundary conditions (Other than the wave boundary and the radiation boundary)	Structures can be set at arbitrary positions on a cell basis Boundary condition can be set at arbitrary position on the surface of the structure Boundary condition type can be selected with input data
Numerical method and algorithm	Discretization	Difference method using staggered mesh Shape approximation using porous model
	Time integration	Euler method Simplified Marker and Cell method (SMAC method)

	Advection term (convection term)	The advection term (convection term) other than VOF function F can be selected from any of the following 1. First order accuracy upwind difference 2. Second-order accuracy central difference 3. DONOR scheme (Hybrid format of (1) and (2) One of the following can be selected as the advection term of the VOF function F 1. Donor · Acceptor method 2. Method considering inclination of interface
	velocity of surface cell setting	Extrapolation (Extrapolate from the flow velocity of two points on the fluid side) Gradient zero (make it the same as flow velocity on fluid side)
	Method of determining the direction of the surface	Adopted NASA-VOF 3D method
	Bubble and water drop	・TimerDoor method
	Solver of simultaneous linear equations	・MILU-BiCGSTAB method
	Control of time step	Fixed input value Automatic time step
Graphic function	Data of specified area and time step (2D only in drawing section)	Flow velocity vector Isolated lines of various physical quantities Free surface with contour lines Filling of existing area of fluid
	Data in chronological order (Use spreadsheet software)	Water level fluctuation from initial water level of designated area Calculated value of specified part, others

Chapter 2 **Analytical model**

In this chapter, we show the analytical model of CADMAS-SURF/3D-MG such as basic equations of three-dimensional incompressible fluids, free surface analysis models and wave-shaping models.

2.1. Coordinate system and difference grid method

The coordinate system is a Cartesian coordinate system, and the analysis area is divided by a rectangular difference cell (see Figure 0-2-1). For the arrangement of each variable, the scalar quantity is the center of the cell, the x direction flow velocity is the center of the cell interface perpendicular to the x axis, the y direction flow velocity is the center of the cell interface perpendicular to the y axis, the z direction flow velocity is perpendicular to the z axis The center of the cell interface, the staggered grid system was adopted (see Figure 0-2-2). For consistency of suffixes and boundary condition processing etc. , virtual cells are provided outside the analysis area.

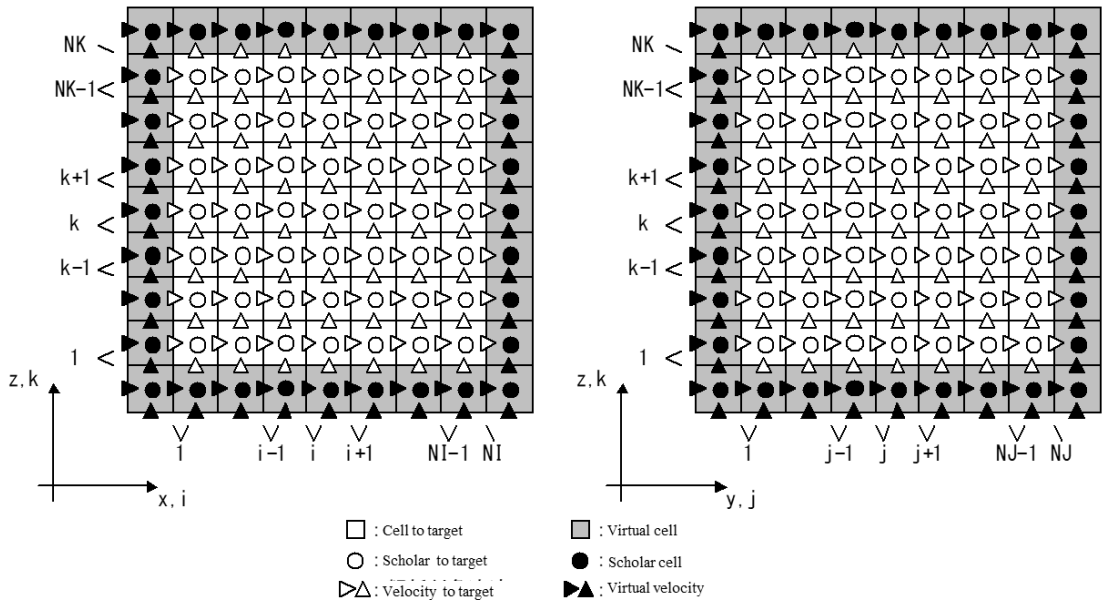


Figure 0-2-1 Coordinate system and difference cell

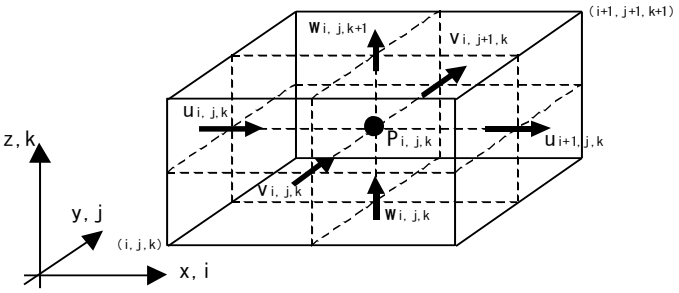


Figure 0-2-2 Variable arrangement in staggered grid system

2.2. Basic equations of three-dimensional incompressible fluid

In the analysis of the wave field in the coastal area, it is indispensable to be able to handle complicated shapes such as seabed slope and permeable wave-eliminating structure. For this reason, the basic equations of CADMAS-SURF/3D-MG include the continuous equation for three-dimensional incompressible viscous fluid and the equation (2. 1) obtained by extending the Navier-Stokes equations based on the porous model[2] (2. 4) was adopted.

・Continuous equation

$$\frac{\partial \gamma_x u}{\partial x}+\frac{\partial \gamma_y v}{\partial y}+\frac{\partial \gamma_z w}{\partial z}=\gamma_v S_\rho \quad \text{(2. 1)}$$

・Navier-Stokes equation

$$\lambda_v\frac{\partial u}{\partial t}+\frac{\partial \lambda_x uu}{\partial x}+\frac{\partial \lambda_y vu}{\partial y}+\frac{\partial \lambda_z wu}{\partial z}=-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial x}+\frac{\partial}{\partial x}\left\{\gamma_x \nu_e\left(2\frac{\partial u}{\partial x}\right)\right\}+\frac{\partial}{\partial y}\left\{\gamma_y \nu_e\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right\}+\frac{\partial}{\partial z}\left\{\gamma_z \nu_e\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right\}-\gamma_v D_x u-R_x+\gamma_v S_u \quad \text{(2. 2)}$$

$$\lambda_v\frac{\partial v}{\partial t}+\frac{\partial \lambda_x uv}{\partial x}+\frac{\partial \lambda_y vv}{\partial y}+\frac{\partial \lambda_z wv}{\partial z}=-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial y}+\frac{\partial}{\partial x}\left\{\gamma_x \nu_e\left(\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\right)\right\}+\frac{\partial}{\partial y}\left\{\gamma_y \nu_e\left(2\frac{\partial v}{\partial y}\right)\right\}+\frac{\partial}{\partial z}\left\{\gamma_z \nu_e\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right\}-\gamma_v D_y v-R_y+\gamma_v S_v \quad \text{(2. 3)}$$

$$\lambda_v\frac{\partial w}{\partial t}+\frac{\partial \lambda_x uw}{\partial x}+\frac{\partial \lambda_y vw}{\partial y}+\frac{\partial \lambda_z ww}{\partial z}=-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial z}+\frac{\partial}{\partial x}\left\{\gamma_x \nu_e\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)\right\}+\frac{\partial}{\partial y}\left\{\gamma_y \nu_e\left(\frac{\partial w}{\partial y}+\frac{\partial v}{\partial z}\right)\right\}+\frac{\partial}{\partial z}\left\{\gamma_z \nu_e\left(2\frac{\partial w}{\partial z}\right)\right\}-\gamma_v D_z w-R_z+\gamma_v S_w-\frac{\gamma_v \rho^* g}{\rho} \quad \text{(2. 4)}$$

Where, t - time, x, y - horizontal coordinates, z - vertical coordinates, u, v, w - component of velocity in the x, y, z direction, ρ - standard fluid density, ρ^* - fruid density considerd bouyancy ν_e - sum of molecular kinematic viscosity coefficient ν and vortical viscosity coefficient ν_t, g - graviyional acceleration, γ_v - porosity ratio , $\gamma_x, \gamma_y, \gamma_z$ - area transmittance in the x, y, z directions. $\lambda_v, \lambda_x, \lambda_y, \lambda_z$ is represented by the following equation with C_M as the

inertial force coefficient, and the second term on the right side is the effect of the inertial force received from the structure.

$$\left.\begin{aligned}\lambda_v &= \gamma_v + (1-\gamma_v)C_M \\ \lambda_x &= \gamma_x + (1-\gamma_x)C_M \\ \lambda_y &= \gamma_y + (1-\gamma_y)C_M \\ \lambda_z &= \gamma_z + (1-\gamma_z)C_M\end{aligned}\right\} \quad \text{(2. 5)}$$

D_x, D_y, D_z are coefficient for energy damping zone, S_ρ, S_u, S_v, S_w are a source term for the wave source which is described after here. Resistance force R_x, R_y, R_z from the porous body are modeled in a form proportional to the square of the flow velocity as follows, using C_D as a resistance coefficient.

$$\left.\begin{aligned}R_x &= \frac{1}{2}\frac{C_D}{\Delta x}(1-\gamma_x)u\sqrt{u^2+v^2+w^2} \\ R_y &= \frac{1}{2}\frac{C_D}{\Delta y}(1-\gamma_y)v\sqrt{u^2+v^2+w^2} \\ R_z &= \frac{1}{2}\frac{C_D}{\Delta z}(1-\gamma_z)w\sqrt{u^2+v^2+w^2}\end{aligned}\right\} \quad \text{(2. 6)}$$

Where, $\Delta x, \Delta y, \Delta z$ are the interval of grids in the x, y, z directions.

2.3. Free surface analysis model

In the free surface analysis model, VOF methods [3) and 4)], which are versatile and capable of analyzing complicated surface geometries, were adopted. The conceptual diagram of free surface modeling by the VOF method is shown in Figure 0-2-3. In the VOF method, the behavior of the free surface is analyzed by using advection equation of the VOF function F obtained by averaging the function expressing "not fluid / not" as the step function per calculation cell and calculating the direction of the surface sequentially. The advection equation of the VOF function F based on the porous model is shown below.

$$\gamma_v \frac{\partial F}{\partial t} + \frac{\partial \gamma_x uF}{\partial x} + \frac{\partial \gamma_y vF}{\partial y} + \frac{\partial \gamma_z wF}{\partial z} = \gamma_v S_F \quad \text{(2. 7)}$$

Where, S_F is a source term for the wave source which is described after here.

As shown in Figure 0-2-4, the VOF function F is a function to express the free surface sharply, unlike the void fraction used in two - phase flow analysis. To discretize the advection equation, a donor-acceptor method specially devised because the surface is not blurred is used. Further, a method considering the inclination of the interface can also be selected. The donor-acceptor method, the method considering the inclination of the interface and the method of determining the direction of the surface are described later.

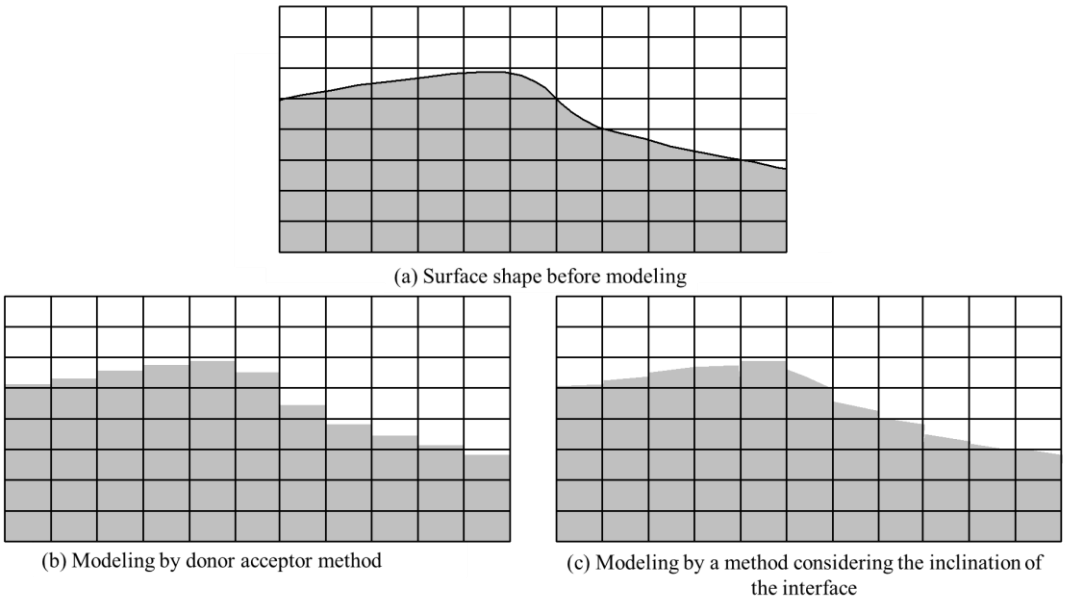


Figure 0-2-3 Modeling of free surface by VOF method

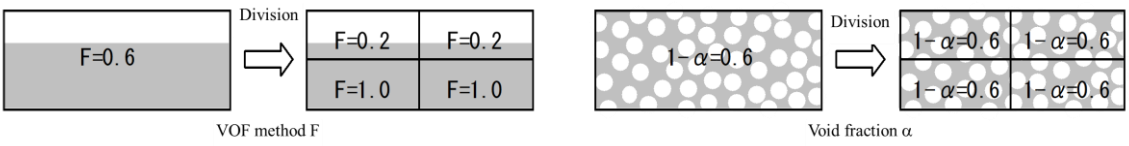


Figure 0-2-4 Difference between VOF function F and void fraction α

2.4. Wave model

(1) Water level and flow velocity for wave generation

In CADMAS-SURF/3D-MG, the following three methods were adopted as a method of calculating the water level and flow velocity of the regular wave.

· Stokes wave fifth approximation solution [5)]

· Knoid wave third order approximation solution [5)]

· Numerical solution by flow function method B [6)]

In the application of Dean's flow function method B, similar to the perturbation solution of stokes wave and the quinoid wave derived based on the second definition of the wave velocity, the physical quantity (water level variation, flow velocity, pressure, Wave velocity) are given. Also, for the purpose of generating irregular waves, we adopted a method of reading data of arbitrary water level and flow velocity (matrix data) created by the user and calculating while interpolating in the time direction and spatial direction.

When trying to generate a wave using the water level $\eta_0(t)$ and the flow velocity $U_0(z,t)$ obtained by the above method, a deviation occurs between the assumed water level $\eta_0(t)$ and the calculated water level $\eta_s(t)$ immediately after the start of wave generation.

Therefore, the following scaling is applied to the flow velocity $U(z,t)$ for wave generation in the x direction or the y direction used for actual calculation so that the inflow / outflow amount at the wave forming position is matched.

$$U(z,t)=U_0\left(z^*,t\right)\cdot\left(\frac{\eta_0+h}{\eta_s+h}\right) \quad \text{(2. 8)}$$

$$z^*=\frac{\eta_0+h}{\eta_s+h}(z+h)-h \quad \text{(2. 9)}$$

where, h is water depth.

In the CADMAS-SURF/3D-MG, as the wave generation model using these wave-generating flow

velocities $U(z,t)$, the following two wave boundaries and wave-generating sources are adopted.

(2) Wave boundary

At the wave boundary, as shown in Figure 0-2-5, the flow velocity $U(z,t)$ for wave formation is set as the flow velocity designation boundary condition in the x direction or the y direction. In order to prevent excessive condition, the VOF function F value and pressure is set the slope zero (actually pressure increment).

(3) Wave source

The source for wave generation[7] is set at the center position ($x = x_s$ or $y = y_s$) of the specified cell (Figure 0-2-5). Since the wave source is not a method of directly designating the flow velocity and the water level, Reflective wave generation can be achieved by combining it with a nonreflective model are described later, which can generats reflected waves.

Only the source terms of the equations (2.1) to (2.4) and (2.7) are shown below.

$$S_\rho = q(z,t) \quad \text{(2. 10)}$$

$$S_u = uq(z,t) \quad \text{(2. 11)}$$

$$S_v = vq(z,t) \quad \text{(2. 12)}$$

$$S_w = wq(z,t) + \frac{\nu}{3}\frac{\partial q(z,t)}{\partial z} \quad \text{(2. 13)}$$

$$S_F = Fq(z,t) \quad \text{(2. 14)}$$

where, $q(z,t)$ is expressed by the following equation with the grid spacing at $x = x_s$ as Δx_s (when wave is generated in the x direction).

$$q(z,t) = 2U(z,t)/\Delta x_s \quad \text{(2. 15)}$$

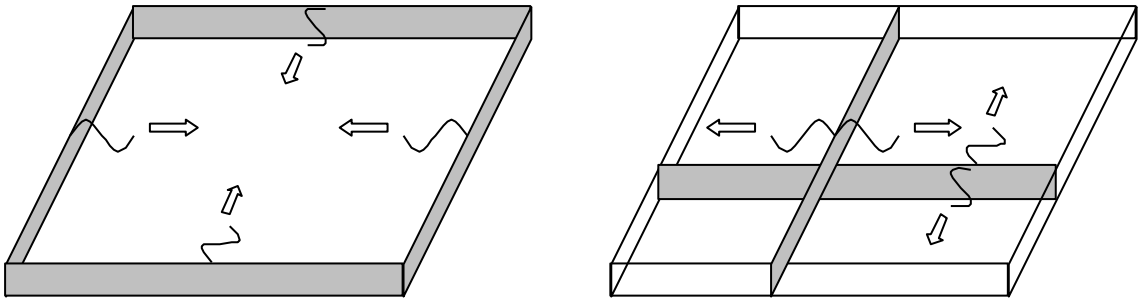


Figure 0-2-5 Wave boundary (left) and wave generating source (right)

2.5. Nonreflective model

In the case of irregular waves, as well as in regular waves, it takes a long time to analyze reflected waves and shallow water deformation etc. of the structure stably, so that it is not artificially set (Nonreflective model) that minimizes the influence of both ends of the analytical region where it does not have to be obtained.

In CADMAS-SURF/3D-MG, the followings were adopted as a nonreflective model.

(1) Radiation boundary of Sommerfeld

Radiation conditions of Sommerfeld shown below are adopted as boundary conditions (see Figure 0-2-6)

$$\frac{\partial f}{\partial t}+C\frac{\partial f}{\partial x}=0 \text{ , or } \frac{\partial f}{\partial t}+C\frac{\partial f}{\partial y}=0 \quad \text{..... (2. 16)}$$

where f is physical quantity such as flow velocity, and C is wave velocity. The wave velocity C is used for the wave velocity of micro amplitude waves.

(2) Energy damping zone

The energy damping zone gradually attenuates the energy of the wave using the region of 1 to 3 wavelengths, which achieves nonreflection (see Figure 0-2-6). Therefore, an extra computation area is required. However, energy damping zone is applicable to various waveforms and has an advantage

that a stable calculation result can be easily obtained.

In the CADMAS-SURF/3D-MG, the damping term[8] proportional to the flow velocity shown below is added to the Navier-Stokes equation of the formula (2. 2) to the equation (2. 4).

・Damping term of x direction flow velocity $= -D_x u$ ..... (2. 17)

$$D_x = \theta_{xy}\sqrt{\frac{g}{h}}(N+1)\left(\frac{\max(|x-x_0|,|y-y_0|)}{l}\right)^N$$

・Damping term of y direction flow velocity $= -D_y v$ ..... (2. 18)

$$D_y = \theta_{xy}\sqrt{\frac{g}{h}}(N+1)\left(\frac{\max(|x-x_0|,|y-y_0|)}{l}\right)^N$$

・Damping term of z direction flow velocity $= -D_z w$ ..... (2. 19)

$$D_z = \theta_z\sqrt{\frac{g}{h}}(N+1)\left(\frac{\max(|x-x_0|,|y-y_0|)}{l}\right)^N$$

where, h is water depth, l and x_0 (or y_0) are the width of energy damping zone and start point, N is a order of distribution function, and θ_{xy} and θ_z are a dimensionless coefficient.

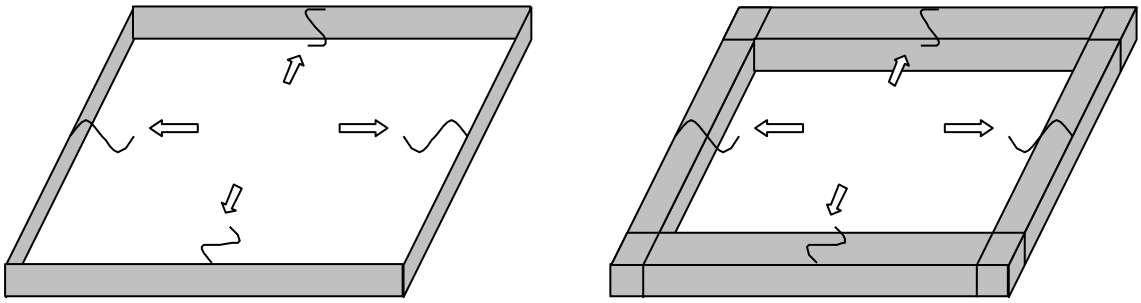


Figure 0-2-6 Sommerfeld radiation boundary (left) and energy damping zone (right) turbulence model

2.6. Turbulent flow model

For the turbulent flow model, we adopted the high Re type k - ε 2 equation model[9] which is highly practical and used in various fields.

High Re type k - ε 2 equation model defines turbulence energy k and turbulent energy dissipation ε as

$$k = \frac{1}{2}\left(u'^2 + v'^2 + w'^2\right) \quad \text{(2. 20)}$$

$$\varepsilon = \nu\left[2\left\{\left(\frac{\partial u'}{\partial x}\right)^2 + \left(\frac{\partial v'}{\partial y}\right)^2 + \left(\frac{\partial w'}{\partial z}\right)^2\right\} + \left(\frac{\partial u'}{\partial y} + \frac{\partial v'}{\partial x}\right)^2 + \left(\frac{\partial v'}{\partial z} + \frac{\partial w'}{\partial y}\right)^2 + \left(\frac{\partial w'}{\partial x} + \frac{\partial u'}{\partial z}\right)^2\right] \quad \text{(2. 21)}$$

by using the fluctuation amount (u', v', w') of the flow velocity, which obtains the solution of advection diffusion equation as below.

$$\gamma_v \frac{\partial k}{\partial t} + \frac{\partial \gamma_x uk}{\partial x} + \frac{\partial \gamma_y vk}{\partial y} + \frac{\partial \gamma_z wk}{\partial z} = \frac{\partial}{\partial x}\left(\gamma_x \nu_k \frac{\partial k}{\partial x}\right) + \frac{\partial}{\partial y}\left(\gamma_y \nu_k \frac{\partial k}{\partial y}\right) + \frac{\partial}{\partial z}\left(\gamma_z \nu_k \frac{\partial k}{\partial z}\right) + \gamma_v\left(G_s + G_T - \varepsilon\right) \quad \text{(2. 22)}$$

$$\gamma_v \frac{\partial \varepsilon}{\partial t} + \frac{\partial \gamma_x u\varepsilon}{\partial x} + \frac{\partial \gamma_y v\varepsilon}{\partial y} + \frac{\partial \gamma_z w\varepsilon}{\partial z} = \frac{\partial}{\partial x}\left(\gamma_x \nu_\varepsilon \frac{\partial \varepsilon}{\partial x}\right) + \frac{\partial}{\partial y}\left(\gamma_y \nu_\varepsilon \frac{\partial \varepsilon}{\partial y}\right) + \frac{\partial}{\partial z}\left(\gamma_z \nu_\varepsilon \frac{\partial \varepsilon}{\partial z}\right) + \gamma_v\left\{C_1 \frac{\varepsilon}{k}\left(G_s + G_T\right)\left(1 + C_3 R_f\right) - C_2 \frac{\varepsilon^2}{k}\right\} \quad \text{(2. 23)}$$

$$G_s = \nu_t\left[2\left\{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial z}\right)^2\right\} + \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right)^2\right] \quad \text{(2. 24)}$$

$$G_T = -\frac{\nu_t}{\rho\sigma_t}\left(g\frac{\partial \rho^*}{\partial z}\right) \quad \text{(2. 25)}$$

$$R_f = \frac{-G_T}{G_s + G_T} \quad \text{(2. 26)}$$

$$\nu_t = C_\mu \frac{k^2}{\varepsilon} \quad \text{(2. 27)}$$

$$\nu_k = \nu + \frac{\nu_t}{\sigma_k} \quad \text{(2. 28)}$$

$$\nu_\varepsilon = \nu + \frac{\nu_t}{\sigma_\varepsilon} \quad \text{(2. 29)}$$

where constants contained by the equation (2. 22) to the equation (2. 29) are obtained by experience, and generally $C_\mu = 0.09$, $\sigma_k = 1.00$, $\sigma_\varepsilon = 1.30$, $C_1 = 1.44$, $C_2 = 1.92$, $C_3 = 0.0$ are adopted. The effect of the turbulence calculated using these equations influences the flow velocity and the pressure by treating the effective kinetic coefficient of the viscous term of the equation of motion as

$$\nu_e = \nu + \nu_t \quad \text{(2. 30)}$$

and

$$p' = p + \frac{2}{3}\rho k \quad \text{(2. 31)}$$

2.7. Energy equation

Assuming that the change in density and constant pressure specific heat is minute, the following energy equation with temperature as the main variable is adopted.

$$\gamma_v \frac{\partial T}{\partial t} + \frac{\partial}{\partial x}\left(\gamma_x u T\right) + \frac{\partial}{\partial y}\left(\gamma_y v T\right) + \frac{\partial}{\partial z}\left(\gamma_z w T\right) = \frac{1}{\rho c_p}\left\{\frac{\partial}{\partial x}\left(\gamma_x \lambda_e \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\gamma_y \lambda_e \frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(\gamma_z \lambda_e \frac{\partial T}{\partial z}\right) + \gamma_v S_T\right\} \quad \text{(2. 32)}$$

$$\lambda_e = \lambda + \frac{\rho c_p \nu_t}{\mathrm{Pr}_t} \quad \text{(2. 33)}$$

$$S_T = \rho c_p T q(z,t) \quad \text{(2. 34)}$$

where, T is temperature, c_p is constant pressure specific heat, λ is thermal conductivity, and Pr_t is turbulent flow prandtl number (input value).

2.8. Concentration transport equations

Assuming that the concentration of the contaminant is very small and the influence on the fluid motion can be neglected, the transport equation of the following concentration was adopted.

$$\gamma_v \frac{\partial c_i}{\partial t} + \frac{\partial}{\partial x}\left(\gamma_x u c_i\right) + \frac{\partial}{\partial y}\left(\gamma_y v c_i\right) + \frac{\partial}{\partial z}\left(\gamma_z w c_i\right) = \frac{\partial}{\partial x}\left(\gamma_x D_{ie} \frac{\partial c_i}{\partial x}\right) + \frac{\partial}{\partial y}\left(\gamma_y D_{ie} \frac{\partial c_i}{\partial y}\right) + \frac{\partial}{\partial z}\left(\gamma_z D_{ie} \frac{\partial c_i}{\partial z}\right) + \gamma_v S_{c_i} \quad \text{(2. 35)}$$

$$D_{ie} = D_i + \frac{\nu_t}{S_{CT}} \quad \text{(2. 36)}$$

$$S_{c_i} = c_i q(z,t) \quad \text{(2. 37)}$$

where c_i is is the concentration (volume fraction) of the i-th component, D_i is the diffusion coefficient of the i-th component, S_{CT} is the turbulent flow Schmidt number (input value).

2.9. Boundary conditions

In CADMAS-SURF/3D-MG, it is possible to use value designation boundaries, logarithmic rule boundaries, etc. in addition to the wave boundary and Sommerfeld boundary described above.

Table 0-2-1 lists the boundary conditions related to flow velocity and pressure when the suffix on the wall surface and on the free surface is shown in Figure 0-2-7, the boundary condition list on the scalar quantity is shown in Table 0-2-2.

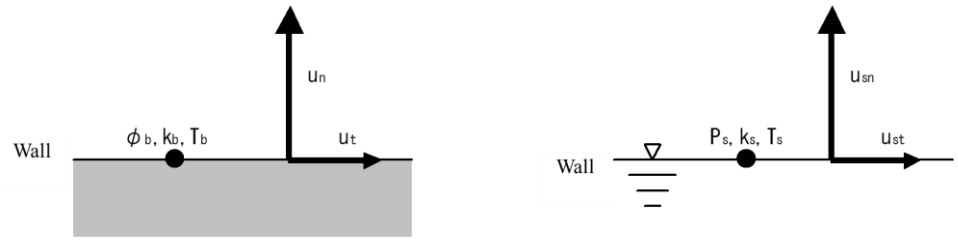


Figure 0-2-7 Suffix on wall and free surface

Table 0-2-1 List of boundary conditions concerning flow velocity and pressure

Boundary condition	Velocity	Pressure
Slip	$u_n = 0$, $\frac{\partial u_t}{\partial n} = 0$	$\frac{\partial \phi_b}{\partial n} = 0$
Non slip	$u_n = 0$, $u_t = 0$	$\frac{\partial \phi_b}{\partial n} = 0$
Value specification (including wave boundary)	$u_n = u_{nin}$, $u_t = u_{tin}$	$\frac{\partial \phi_b}{\partial n} = 0$
Free	$\frac{\partial u_n}{\partial n} = 0$ (Corrected after calculation of potential function), $\frac{\partial u_t}{\partial n} = 0$	$\phi_b = 0$
Radiation boundary	$\frac{\partial u_n}{\partial t} + C\frac{\partial u_n}{\partial n} = 0$, $\frac{\partial u_t}{\partial t} + C\frac{\partial u_t}{\partial n} = 0$ The wave velocity C is obtained from a minute amplitude wave.	$\frac{\partial \phi_b}{\partial n} = 0$
Logarithmic law	$u_n = 0$, $\frac{u}{u^*} = \frac{1}{\kappa}\log\left(\frac{u^* y}{\nu}\right) + A$, $u^* = \sqrt{\tau/\rho} =$ Friction speed $\kappa = 0.4$, $A = 5.5$ $y =$ Distance from the wall, $u =$ Tangential flow velocity (position of y)	$\frac{\partial \phi_b}{\partial n} = 0$
Perfect rough surface	$u_n = 0$, $\frac{u}{u^*} = \frac{1}{\kappa}\log\left(\frac{y}{k_s}\right) + A + 3.0$, $u^* = \sqrt{\tau/\rho} =$ Friction speed $k_s =$ Wall surface roughness, $\kappa = 0.4$, $A = 5.5$ $y =$ Distance from the wall, $u =$ Tangential flow	$\frac{\partial \phi_b}{\partial n} = 0$

	velocity (position)	
Free surface boundary	For the normal direction, it is calculated as appropriate from the equation of continuity, zero gradient, and extrapolation. For tangential direction, it is calculated as appropriate from zero gradient, and extrapolation.	$p_s = p_{GAS}$

Note) The boundary condition of the pressure on the wall is expressed as the boundary condition of the potential function φ.

Note) The suffix (in) is the input value.

Table 0-2-2 List of boundary conditions on scalar quantity

Boundary condition	Turbulent flow	Temperature	Concentration
Value specification	$k_b = k_{in}$, $\varepsilon_b = \varepsilon_{in}$	$T_b = T_{in}$	$c_{ib} = c_{iin}$
Free (zero gradient, adiabatic)	$\frac{\partial k_b}{\partial n} = 0$, $\frac{\partial \varepsilon_b}{\partial n} = 0$	$\frac{\partial T_b}{\partial n} = 0$	$\frac{\partial c_{ib}}{\partial n} = 0$
Heat flux (diffusion flux)	—	$-\lambda_e \left.\frac{\partial T}{\partial n}\right\|_b = q_{in}$	$-D_{ie} \left.\frac{\partial c_i}{\partial n}\right\|_b = q_{in}$
Heat transfer (mass transfer)	—	$-\lambda_e \left.\frac{\partial T}{\partial n}\right\|_b = h(T_b - T_\infty)$ h is the heat transfer coefficient. T_∞ is external temperature.	$-D_{ie} \left.\frac{\partial c_i}{\partial n}\right\|_b = k_i(c_{ib} - c_{i\infty})$ k_i is the mass transfer .coefficient $c_{i\infty}$ is external concentration.
Logarithmic law and complete rough surface	$k_c = \frac{(u^*)^2}{\sqrt{C_\mu}}$, $\varepsilon_c = \frac{(u^*)^3}{\kappa y}$ k_c , ε_c are the value of the cell in contact with the interface	—	—
Emission boundary	$\frac{\partial k_b}{\partial t} + C\frac{\partial k_b}{\partial n} = 0$ $\frac{\partial \varepsilon_b}{\partial t} + C\frac{\partial \varepsilon_b}{\partial n} = 0$	$\frac{\partial T_b}{\partial t} + C\frac{\partial T_b}{\partial n} = 0$	$\frac{\partial c_{ib}}{\partial t} + C\frac{\partial c_{ib}}{\partial n} = 0$
Free surface boundary	$\frac{\partial k_s}{\partial n} = 0$, $\frac{\partial \varepsilon_s}{\partial n} = 0$	$\frac{\partial T_s}{\partial n} = 0$	$\frac{\partial c_{is}}{\partial n} = 0$

Note) The suffix (in) is the input value

Chapter 3 **Numerical solution**

This chapter describes numerical solutions[10), 11)] of CADMAS-SURF/3D-MG such as discretization in the time direction, discretization in the spatial direction and solving simultaneous linear equations.

3.1. Discretization policy

In development of CADMAS-SURF/3D-MG, the policy of discretization is set as follows.

(1) The staggered grid system shown in Section 2.1 is adopted.

(2) A variable grid width that allows the grid width to be freely set is adopted.

(3) The basic equations described in the preservation format are discretized by the control volume method.

(4) The difference in the spatial direction other than the advection term is the central difference of the second order accuracy, and the difference other than the advection term is the upwind difference etc.

(5) Linear interpolation and area interpolation are used for the interpolation method.

(6) Discretization in the time direction is based on the Euler method, and SMAC method (Simplified Marker and Cell[12)]) implicitly evaluating the pressure term is adopted for coupling of the equation of motion and continuous equation.

3.2. Time discretization

For the discretization in the time direction for calculating the physical quantity f^{n+1} at a new time $t+\Delta t$ from the physical quantity f^{n} at a certain time t, the Euler method (explicit method) is used. The SMAC method which implicitly evaluates the pressure term is adopted for the coupling of

the equation of motion and the continuous equation.The conceptual diagram of discretization in the time direction is shown in Figure 0-3-1.

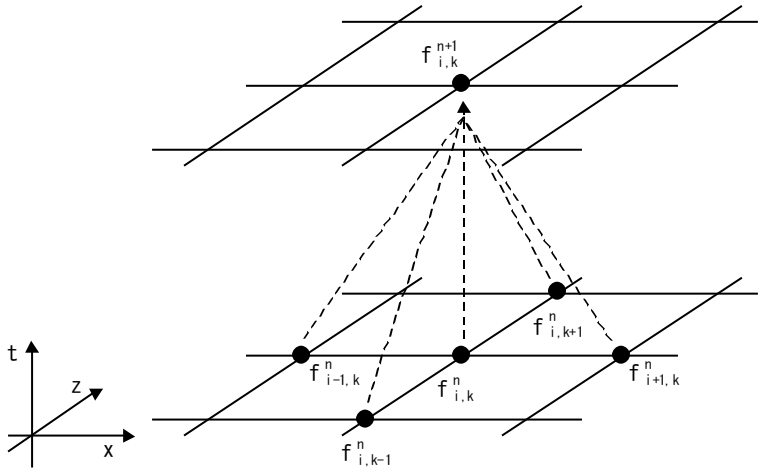


Figure 0-3-1 Concept of time discretization

3.2.1. Euler method

The Euler method is the most fundamental explicit method. When discretizing the basic equation as

$$\frac{\partial f}{\partial t} = G(f) \quad \cdots\cdots (3.1)$$

, it becomes the following equation.

$$f^{n+1} = f^{n} + \Delta t G(f^{n}) \quad \cdots\cdots (3.2)$$

3.2.2. SMAC method

Navier-Stokes equations and continuous equations have no pressure fluctuation term. In the SMAC method, the pressure term is implicitly evaluated to obtain the pressure at the new time. When discretizing the basic equations of expressions (2. 1) to (2. 4) in the time direction by the SMAC method,

$$\frac{\partial \gamma_x u^{n+1}}{\partial x} + \frac{\partial \gamma_y v^{n+1}}{\partial y} + \frac{\partial \gamma_z w^{n+1}}{\partial z} = \gamma_v S_\rho \quad \cdots\cdots (3.3)$$

$$u^{n+1} = u^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n+1}}{\partial x} + G_x\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \cdots\cdots (3.4)$$

$$v^{n+1} = v^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n+1}}{\partial y} + G_y\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \cdots\cdots (3.5)$$

$$w^{n+1} = w^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n+1}}{\partial z} + G_z\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \cdots\cdots (3.6)\ .$$

In addition, the terms other than the time term and the pressure term of expressions (2. 2) to (2. 4) are collectively shown as $G_x(u^n, v^n, w^n)$ and $G_z(u^n, v^n, w^n)$ respectively.

$$u^{\sim} = u^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n}}{\partial x} + G_x\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \cdots\cdots (3.7)$$

$$v^{\sim} = v^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n}}{\partial y} + G_y\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \cdots\cdots (3.8)$$

$$w^{\sim} = w^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n}}{\partial z} + G_z\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \cdots\cdots (3.9)$$

$$\phi = -\frac{\Delta t}{\rho}\left(P^{n+1} - P^{n}\right) \quad \text{(3. 10)}$$

where temporary flow rate are $u^{\sim}$, $v^{\sim}$, $w^{\sim}$ and potential function is ϕ which is rearranged equation (3.6) from equation (3. 4) as

$$u^{n+1} - u^{\sim} = \frac{\Delta t}{\lambda_v}\left(-\frac{\gamma_v}{\rho}\frac{\partial P^{n+1} - P^{n}}{\partial x}\right) = \frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial x} \quad \text{(3. 11)}$$

$$v^{n+1} - v^{\sim} = \frac{\Delta t}{\lambda_v}\left(-\frac{\gamma_v}{\rho}\frac{\partial P^{n+1} - P^{n}}{\partial y}\right) = \frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial y} \quad \text{(3. 12)}$$

$$w^{n+1} - w^{\sim} = \frac{\Delta t}{\lambda_v}\left(-\frac{\gamma_v}{\rho}\frac{\partial P^{n+1} - P^{n}}{\partial z}\right) = \frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial z} \quad \text{(3. 13)}$$

By multiplying $\frac{\partial}{\partial x}\gamma_x$, $\frac{\partial}{\partial y}\gamma_y$ and $\frac{\partial}{\partial z}\gamma_z$ from expression (3.11) to (3.13), respectively, and taking th sum, it becomes

$$\begin{aligned}&\frac{\partial}{\partial x}\left(\gamma_x\frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(\gamma_y\frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial y}\right) + \frac{\partial}{\partial z}\left(\gamma_z\frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial z}\right)\\ &= \left(\frac{\partial \gamma_x u^{n+1}}{\partial x} + \frac{\partial \gamma_v v^{n+1}}{\partial y} + \frac{\partial \gamma_z w^{n+1}}{\partial z}\right) - \left(\frac{\partial \gamma_x u^{\sim}}{\partial x} + \frac{\partial \gamma_y v^{\sim}}{\partial y} + \frac{\partial \gamma_z w^{\sim}}{\partial z}\right)\end{aligned} \quad \text{(3. 14)}$$

where by substituting the expression (3. 3) into the first term on the right side of the expression (3. 14), the following SMAC algorithm algorithm is obtained.

(1) Calculates temporary flow velocity （$u^{\sim}$, $v^{\sim}$, $w^{\sim}$） using the following equation.

$$u^{\sim} = u^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n}}{\partial x} + G_x\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \text{(3. 15)}$$

$$v^{\sim} = v^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n}}{\partial y} + G_y\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \text{(3. 16)}$$

$$w^{\sim} = w^{n} + \frac{\Delta t}{\lambda_v}\left\{-\frac{\gamma_v}{\rho}\frac{\partial P^{n}}{\partial z} + G_z\left(u^{n}, v^{n}, w^{n}\right)\right\} \quad \text{(3. 17)}$$

(2) Solves the following Poisson equation and find a potential function.

$$\frac{\partial}{\partial x}\left(\gamma_x\frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(\gamma_y\frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial y}\right) + \frac{\partial}{\partial z}\left(\gamma_z\frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial z}\right) = \gamma_v S_\rho - \left(\frac{\partial \gamma_x u^{\sim}}{\partial x} + \frac{\partial \gamma_y v^{\sim}}{\partial y} + \frac{\partial \gamma_z w^{\sim}}{\partial z}\right) \quad \text{(3. 18)}$$

(3) Obtains u^{n+1}, v^{n+1}, w^{n+1} and P^{n+1} at the new time using the following equation and repeat the steps 1 to 3 until the required time.

$$u^{n+1} = u^{\sim} + \frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial x} \quad \text{(3. 19)}$$

$$v^{n+1} = v^{\sim} + \frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial y} \quad \text{(3. 20)}$$

$$w^{n+1} = w^{\sim} + \frac{\gamma_v}{\lambda_v}\frac{\partial \phi}{\partial z} \quad \text{(3. 21)}$$

$$P^{n+1} = P^{n} - \frac{\rho}{\Delta t}\phi \quad \text{(3. 22)}$$

3.3. Discretization in spatial direction

3.3.1. Discretization policy

Up to this paragraph, we have described the known physical quantity as f^{n} and the physical quantity at the new time as f^{n+1} by using n, $n+1$ as the suffix in the time direction. However, for the sake of simplicity, the known physical quantity is described here after.

In the discretization in the spatial direction, as shown in Figure 0-3-2, a minute amount in the spatial direction is defined as

$$\Delta x_i = x_{i+1} - x_i \quad \text{(3. 23)}$$

$$\Delta y_j = y_{j+1} - y_j \quad \text{(3. 24)}$$

$$\Delta z_k = z_{k+1} - z_k \quad \text{(3. 25)}$$

$$\delta x_i = \frac{\Delta x_{i-1} + \Delta x_i}{2} \quad \text{(3. 26)}$$

$$\delta y_j = \frac{\Delta y_{j-1} + \Delta y_j}{2} \quad \text{(3. 27)}$$

$$\delta z_k = \frac{\Delta z_{k-1} + \Delta z_k}{2} \quad \text{(3. 28)}$$

where the amount with Δ is the width of the cell and the value with δ is the width between cell center. For simplification of the discrete expression, the variable f to be interpolated is surrounded by $\{\ \}$ and $\lfloor\ \rfloor$, depending on the suffix, which is described as

$$\{f\}^{\Delta x}_{i,j,k} = \frac{\Delta x_{i-1}}{\Delta x_{i-1}+\Delta x_i} f_{i,j,k} + \frac{\Delta x_i}{\Delta x_{i-1}+\Delta x_i} f_{i-1,j,k} \quad \text{(3. 29)}$$

$$\lfloor f \rfloor^{\Delta x}_{i,j,k} = \frac{\Delta x_i}{\Delta x_{i-1}+\Delta x_i} f_{i,j,k} + \frac{\Delta x_{i-1}}{\Delta x_{i-1}+\Delta x_i} f_{i-1,j,k} \quad \text{(3. 30)}$$

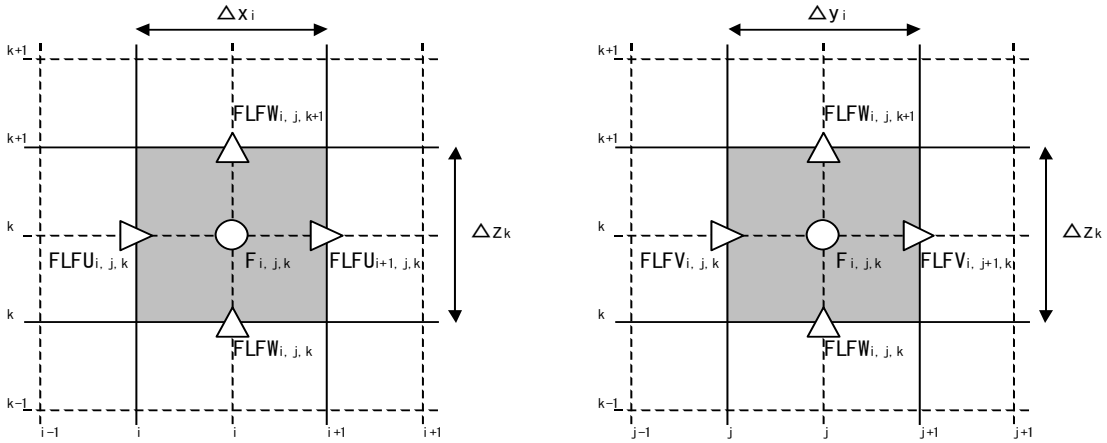
$$\{f\}^{\delta x}_{i,j,k} = \lfloor f \rfloor^{\delta x}_{i,j,k} = \frac{1}{2} f_{i+1,j,k} + \frac{1}{2} f_{i,j,k} \quad \text{(3. 31)}$$

where the equation (3. 29) corresponds to the case where the value defined at the cell center is linearly interpolated to the cell interface, the equation (3. 30) corresponds to the case where the value defined at the cell center is interpolated to the cell interface and the equation (3.31) corresponds to the case where the value defined at the cell interface is linearly interpolated (the area interpolation is also the same) at the cell center. For example, using the description of equation (3. 29), when linearly interpolating the value defined at the cell center to the grid point on the xz plane, it can be described using double as follows. For example, using the description of equation (3. 29), when linearly interpolating the value defined at the cell center to the grid point on the xz plane, it can be described using double $\{\ \}$ as follows.

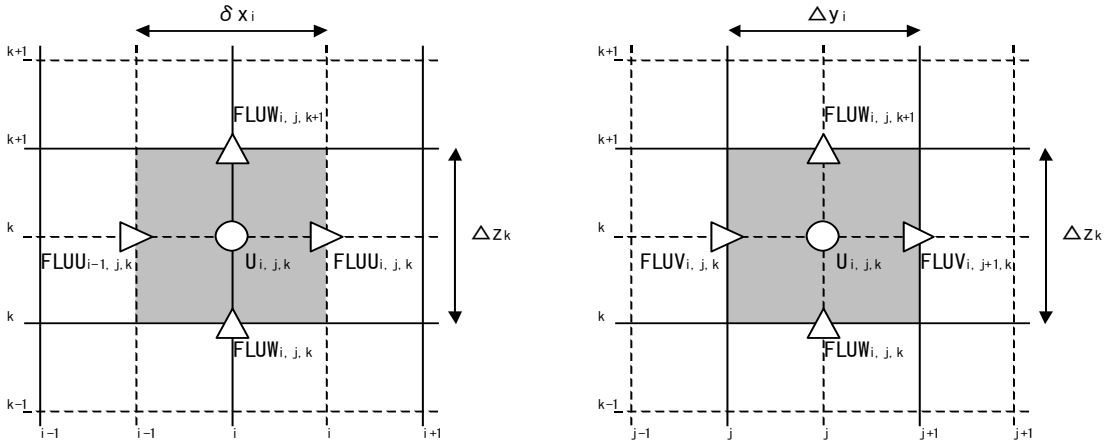
$$\begin{aligned}\{\{f\}^{\Delta x}_{i,j,k}\}^{\Delta z}_{i,j,k} = {} & \frac{\Delta x_{i-1}}{\Delta x_{i-1}+\Delta x_i}\frac{\Delta z_{k-1}}{\Delta z_{k-1}+\Delta z_k} f_{i,j,k} + \frac{\Delta x_i}{\Delta x_{i-1}+\Delta x_i}\frac{\Delta z_{k-1}}{\Delta z_{k-1}+\Delta z_k} f_{i-1,j,k} \\ & + \frac{\Delta x_{i-1}}{\Delta x_{i-1}+\Delta x_i}\frac{\Delta z_k}{\Delta z_{k-1}+\Delta z_k} f_{i,j,k-1} + \frac{\Delta x_i}{\Delta x_{i-1}+\Delta x_i}\frac{\Delta z_k}{\Delta z_{k-1}+\Delta z_k} f_{i-1,j,k-1}\end{aligned} \quad \text{(3. 32)}$$

In addition, the following description for the primary upwind difference is also prepared.

$$\langle c|a,b\rangle = \begin{cases} ca : (c \geq 0) \\ cb : (c < 0) \end{cases} \quad \text{(3. 33)}$$



(a) Scalar quantity



(b) x direction flow velocity

Figure 0-3-2 Minute amount in space direction and control volume

3.3.2. Transformation based on control volume method

In the control volume method [13], the control volume is set around the physical quantity ($u_{i,j,k}$, $v_{i,j,k}$, $w_{i,j,k}$, $F_{i,j,k}$) to be obtained. The amount of inflow and outflow through these interfaces and the amount to be generated and annihilated in the control volume are determined. In the staggered grid system, as shown in Figure III - 3 - 2, the setting position of the control volume is different according

to the physical quantity. In order to perform discretization in the spatial direction by the control volume method, we transform the equation (2.7) discretized from equations (3. 15) to (3.17) and Euler's method as follows.

$$\begin{aligned} u^{\sim} = u &+ \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial x}\left\{-\lambda_x uu + \gamma_x \nu_e\left(2\frac{\partial u}{\partial x}\right)\right\} + \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial y}\left\{-\lambda_y vu + \gamma_y \nu_e\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right\} \\ &+ \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial z}\left\{-\lambda_z wu + \gamma_z \nu_e\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right\} + \frac{\Delta t}{\lambda_v}\left(-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial x} - \gamma_v D_x u - R_x + \gamma_v S_u\right) \end{aligned} \quad \text{(3. 34)}$$

$$\begin{aligned} v^{\sim} = v &+ \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial x}\left\{-\lambda_x uv + \gamma_x \nu_e\left(\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\right)\right\} + \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial y}\left\{-\lambda_y vv + \gamma_y \nu_e\left(2\frac{\partial v}{\partial y}\right)\right\} \\ &+ \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial z}\left\{-\lambda_z wv + \gamma_z \nu_e\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right\} + \frac{\Delta t}{\lambda_v}\left(-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial y} - \gamma_v D_y v - R_y + \gamma_v S_v\right) \end{aligned} \quad \text{(3. 35)}$$

$$\begin{aligned} w^{\sim} = w &+ \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial x}\left\{-\lambda_x uw + \gamma_x \nu_e\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)\right\} + \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial y}\left\{-\lambda_y vw + \gamma_y \nu_e\left(\frac{\partial w}{\partial y}+\frac{\partial v}{\partial z}\right)\right\} \\ &+ \frac{\Delta t}{\lambda_v}\frac{\partial}{\partial z}\left\{-\lambda_z ww + \gamma_z \nu_e\left(2\frac{\partial w}{\partial z}\right)\right\} + \frac{\Delta t}{\lambda_v}\left(-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial z} - \gamma_v D_z w - R_z + \gamma_v S_w - \frac{\gamma_v \rho^* g}{\rho}\right) \end{aligned} \quad \text{(3. 36)}$$

$$F^{n+1} = F + \frac{\Delta t}{\gamma_v}\frac{\partial}{\partial x}\left(-\gamma_x uF\right) + \frac{\Delta t}{\gamma_v}\frac{\partial}{\partial y}\left(-\gamma_y vF\right) + \frac{\Delta t}{\gamma_v}\frac{\partial}{\partial z}\left(-\gamma_z wF\right) + \Delta t S_F \quad \text{(3. 37)}$$

where the amount of the second term to the fourth term on the right side of each equation flowing in and out through the interface, and the fifth term is the amount of generation and extinction in the control volume.

3.3.3. Navier-Stokes equation: x direction flow velocity

The amount of inlet and outlet through the interface is discribed by the symbol in Figure 0-3-2 and the amount to be generated and extinguished in the control volume is described by $QU_{i,j,k}$, and equation (3. 34) is discretized as follows.

$$u^{\sim} = u + \frac{\Delta t}{\lambda_v \rfloor_{i,j,k}^{\Delta x}}\left\{\begin{aligned} &\frac{1}{\delta x_i}\left(FLUU_{i,j,k} - FLUU_{i-1,j,k}\right) + \frac{1}{\Delta y_j}\left(FLUV_{i,j+1,k} - FLUV_{i,j,k}\right) \\ &+ \frac{1}{\Delta z_k}\left(FLUW_{i,j,k+1} - FLUW_{i,j,k}\right) + QU_{i,j,k} \end{aligned}\right\} \quad \text{(3. 38)}$$

The terms of (i, j, k) are shown below.

(1) $FLUU_{i,j,k} = \left\{ -\lambda_x uu + \gamma_x \nu_e \left(2\frac{\partial u}{\partial x} \right) \right\}_{i,j,k}$

$$\left(\lambda_x uu\right)_{i,j,k} = \left\langle \{\lambda_x u\}_{i,j,k}^{\delta x} \middle| u_{i,j,k}, u_{i+1,j,k} \right\rangle \quad \text{(3. 39)}$$

$$\left\{ \gamma_x \nu_e \left(2\frac{\partial u}{\partial x} \right) \right\}_{i,j,k} = \{\gamma_x\}_{i,j,k}^{\delta x} \nu_{e.i,j,k} \left[2\frac{u_{i+1,j,k} - u_{i,j,k}}{\Delta x_i} \right] \quad \text{(3. 40)}$$

(2) $FLUV_{i,j,k} = \left\{ -\lambda_y vu + \gamma_y \nu_e \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\}_{i,j,k}$

$$\left(\lambda_y vu\right)_{i,j,k} = \left\langle \lfloor \lambda_y v \rfloor_{i,j,k}^{\Delta x} \middle| u_{i,j-1,k}, u_{i,j,k} \right\rangle \quad \text{(3. 41)}$$

$$\left\{ \gamma_y \nu_e \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\}_{i,j,k} = \lfloor \gamma_y \rfloor_{i,j,k}^{\Delta x} \{\{\nu_e\}_{i,j,k}^{\Delta x}\}_{i,j,k}^{\Delta y} \left[\frac{u_{i,j,k} - u_{i,j-1,k}}{\delta y_j} + \frac{v_{i,j,k} - v_{i-1,j,k}}{\delta x_i} \right] \quad \text{(3. 42)}$$

(3) $FLUW_{i,j,k} = \left\{ -\lambda_z wu + \gamma_z \nu_e \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right\}_{i,j,k}$

$$\left(\lambda_z wu\right)_{i,j,k} = \left\langle \lfloor \lambda_z w \rfloor_{i,j,k}^{\Delta x} \middle| u_{i,j,k-1}, u_{i,j,k} \right\rangle \quad \text{(3. 43)}$$

$$\left\{ \gamma_z \nu_e \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right\}_{i,j,k} = \lfloor \gamma_y \rfloor_{i,j,k}^{\Delta x} \{\{\nu_e\}_{i,j,k}^{\Delta x}\}_{i,j,k}^{\Delta z} \left[\frac{u_{i,j,k} - u_{i,j,k-1}}{\delta z_k} + \frac{w_{i,j,k} - w_{i-1,j,k}}{\delta x_i} \right] \quad \text{(3. 44)}$$

(4) $QU_{i,j,k} = \left(-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial x} - \gamma_v D_x u - R_x + \gamma_v S_u \right)_{i,j,k}$

$$\left(\frac{\gamma_v}{\rho}\frac{\partial p}{\partial x} \right)_{i,j,k} = \frac{\lfloor \gamma_v \rfloor_{i,j,k}^{\Delta x}}{\rho} \left[\frac{P_{i,j,k} - P_{i-1,j,k}}{\delta x_i} \right] \quad \text{(3. 45)}$$

$$\left(\gamma_v D_x u\right)_{i,j,k} = \lfloor \gamma_v \rfloor_{i,j,k}^{\Delta x} D_{xi,j,k} u_{i,j,k}^{n+1} \quad \text{(3. 46)}$$

$$\left(R_x\right)_{i,j,k} = \frac{1}{2} \frac{\lfloor C_D \rfloor_{i,j,k}^{\Delta x}}{\delta x_i} \left(1 - \gamma_{xi,j,k}\right) u_{i,j,k} \sqrt{u_{i,j,k}^2 + \left(\{\{v\}_{i,j,k}^{\delta y}\}_{i,j,k}^{\Delta x} \right)^2 + \left(\{\{w\}_{i,j,k}^{\delta z}\}_{i,j,k}^{\Delta x} \right)^2} \quad \text{(3. 47)}$$

$$\left(\gamma_v S_u\right)_{i,j,k} = \left\lfloor \gamma_v \right\rfloor^{\Delta x}_{i,j,k} u_{i,j,k} q_k \frac{\Delta x_s}{2\delta x_i} \quad \text{(3. 48)}$$

Here, we decided to handle expressions (3. 46) implicitly so that the artificially set energy damping term does not make the calculation unstable. Since the equation (3. 48) is calculated on the left and right of the cell in which the wave source is set, the source amount is distributed according to the grid interval.

3.3.4. Navier-Stokes equation: y direction flow velocity

The amount of inlet and outlet through the interface is described by $FLVU_{i,j,k}$ and $FLVW_{i,j,k}$ like the symbols in Figure 0-3-2, and the amount to be generated and extinguished in the control volume is written as $QV_{i,j,k}$, and the equation (3. 35) is discretized as follows.

$$v^{\sim} = v + \frac{\Delta t}{\left\lfloor \lambda_v \right\rfloor^{\Delta y}_{i,j,k}} \left\{ \begin{array}{l} \frac{1}{\Delta x_i}\left(FLVU_{i+1,j,k} - FLVU_{i,j,k}\right) + \frac{1}{\delta y_j}\left(FLVV_{i,j,k} - FLVV_{i,j-1,k}\right) \\ + \frac{1}{\Delta z_k}\left(FLVW_{i,j,k+1} - FLVW_{i,j,k}\right) + QV_{i,j,k} \end{array} \right\} \quad \text{(3. 49)}$$

The terms of (i, j, k) are shown below.

(1) $FLVU_{i,j,k} = \left\{ -\lambda_x uv + \gamma_x \nu_e \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) \right\}_{i,j,k}$

$$\left(\lambda_x uv\right)_{i,j,k} = \left\langle \left\lfloor \lambda_x u \right\rfloor^{\Delta x}_{i,j,k} \middle| v_{i-1,j,k}, v_{i,j,k} \right\rangle \quad \text{(3. 50)}$$

$$\left\{ \gamma_x \nu_e \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) \right\}_{i,j,k} = \left\lfloor \gamma_x \right\rfloor^{\Delta x}_{i,j,k} \left\{ \left\{ \nu_e \right\}^{\Delta x}_{i,j,k} \right\}^{\Delta y}_{i,j,k} \left[\frac{v_{i,j,k} - v_{i-1,j,k}}{\delta x_i} + \frac{u_{i,j,k} - u_{i,j-1,k}}{\delta y_i} \right] \quad \text{(3. 51)}$$

(2) $FLVV_{i,j,k} = \left\{ -\lambda_y vv + \gamma_y \nu_e \left(2\frac{\partial v}{\partial y} \right) \right\}_{i,j,k}$

$$\left(\lambda_y vv\right)_{i,j,k} = \left\langle \left\{ \lambda_y v \right\}^{\delta y}_{i,j,k} \middle| v_{i,j,k}, v_{i,j+1,k} \right\rangle \quad \text{(3. 52)}$$

$$\left\{\gamma_y \nu_e\left(2\frac{\partial v}{\partial y}\right)\right\}_{i,j,k} = \left\{\gamma_y\right\}^{\delta y}_{i,j,k} \nu_{e.i,j,k}\left[2\frac{v_{i,j+1,k}-v_{i,j,k}}{\Delta y_j}\right] \quad \text{(3. 53)}$$

(3) $FLVW_{i,j,k} = \left\{-\lambda_z wv + \gamma_z \nu_e\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right\}_{i,j,k}$

$$\left(\lambda_z wv\right)_{i,j,k} = \left\langle \lfloor \lambda_z w \rfloor^{\Delta z}_{i,j,k} \middle| v_{i,j,k-1}, v_{i,j,k} \right\rangle \quad \text{(3. 54)}$$

$$\left\{\gamma_z \nu_e\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right\}_{i,j,k} = \lfloor \gamma_z \rfloor^{\Delta z}_{i,j,k} \left\{\left\{\nu_e\right\}^{\Delta x}_{i,j,k}\right\}^{\Delta z}_{i,j,k}\left[\frac{v_{i,j,k}-v_{i,j,k-1}}{\delta z_k}+\frac{w_{i,j,k}-w_{i,j-1,k}}{\delta y_i}\right] \quad \text{(3. 55)}$$

(4) $QV_{i,j,k} = \left(-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial y} - \gamma_v D_y v - R_y + \gamma_v S_v\right)_{i,j,k}$

$$\left(\frac{\gamma_v}{\rho}\frac{\partial p}{\partial y}\right)_{i,j,k} = \frac{\lfloor \gamma_v \rfloor^{\Delta y}_{i,j,k}}{\rho}\left[\frac{P_{i,j,k}-P_{i,j-1,k}}{\delta y_j}\right] \quad \text{(3. 56)}$$

$$\left(\gamma_v D_y v\right)_{i,j,k} = \lfloor \gamma_v \rfloor^{\Delta y}_{i,j,k} D_{yi,j,k} v^{n+1}_{i,j,k} \quad \text{(3. 57)}$$

$$\left(R_y\right)_{i,j,k} = \frac{1}{2}\frac{\lfloor C_D \rfloor^{\Delta y}_{i,j,k}}{\delta y_j}\left(1-\gamma_{yi,j,k}\right)v_{i,j,k}\sqrt{\left(\left\{\left\{u\right\}^{\delta x}_{i,j,k}\right\}^{\Delta y}_{i,j,k}\right)^2 + v^2_{i,j,k} + \left(\left\{\left\{w\right\}^{\delta z}_{i,j,k}\right\}^{\Delta y}_{i,j,k}\right)^2} \quad \text{(3. 58)}$$

$$\left(\gamma_v S_v\right)_{i,j,k} = \lfloor \gamma_v \rfloor^{\Delta y}_{i,j,k} v_{i,j,k} q_k \frac{\Delta y_s}{2\delta y_j} \quad \text{(3. 59)}$$

3.3.5. Navier-Stokes equation: z direction flow velocity

The amount of inlet and outlet through the interface is described by $FLWU_{i,j,k}$ and $FLWV_{i,j,k}$ like the symbols in Figure 0-3-2, and the amount to be generated and extinguished in the control volume is written as $QW_{i,j,k}$, and the equation (3. 36) is discretized as follows.

$$w^{\sim} = w + \frac{\Delta t}{\left\lfloor \lambda_v \right\rfloor_{i,j,k}^{\Delta z}} \left\{ \begin{array}{l} \dfrac{1}{\Delta x_i}\left(FLWU_{i+1,j,k} - FLWU_{i,j,k}\right) + \dfrac{1}{\Delta y_j}\left(FLWV_{i,j+1,k} - FLWV_{i,j,k}\right) \\ + \dfrac{1}{\delta z_k}\left(FLWW_{i,j,k} - FLWW_{i,j,k-1}\right) + QW_{i,j,k} \end{array} \right\} \quad \cdots\cdots (3.60)$$

The terms of (i, j, k) are shown below.

(1) $$FLWU_{i,j,k} = \left\{ -\lambda_x uw + \gamma_x \nu_e \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right\}_{i,j,k}$$

$$\left(\lambda_x uw\right)_{i,j,k} = \left\langle \left\lfloor \lambda_x u \right\rfloor_{i,j,k}^{\Delta x} \middle| w_{i-1,j,k}, w_{i,j,k} \right\rangle \quad \cdots\cdots (3.61)$$

$$\left\{ \gamma_x \nu_e \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right\}_{i,j,k} = \left\lfloor \gamma_x \right\rfloor_{i,j,k}^{\Delta x} \left\{ \nu_e \right\}_{i,j,k}^{\Delta x} \left\{ \right\}_{i,j,k}^{\Delta z} \left[\frac{w_{i,j,k} - w_{i-1,j,k}}{\delta x_i} + \frac{u_{i,j,k} - u_{i,j,k-1}}{\delta z_k} \right] \quad \cdots\cdots (3.62)$$

(2) $$FLWV_{i,j,k} = \left\{ -\lambda_y vw + \gamma_y \nu_e \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) \right\}_{i,j,k}$$

$$\left(\lambda_y vw\right)_{i,j,k} = \left\langle \left\lfloor \lambda_y v \right\rfloor_{i,j,k}^{\Delta y} \middle| w_{i,j-1,k}, w_{i,j,k} \right\rangle \quad \cdots\cdots (3.63)$$

$$\left\{ \gamma_y \nu_e \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) \right\}_{i,j,k} = \left\lfloor \gamma_y \right\rfloor_{i,j,k}^{\Delta y} \left\{ \nu_e \right\}_{i,j,k}^{\Delta y} \left\{ \right\}_{i,j,k}^{\Delta z} \left[\frac{w_{i,j,k} - w_{i,j-1,k}}{\delta y_j} + \frac{v_{i,j,k} - v_{i,j,k-1}}{\delta z_k} \right] \quad \cdots\cdots (3.64)$$

(3) $$FLWW_{i,j,k} = \left\{ -\lambda_z ww + \gamma_z \nu_e \left(2\frac{\partial w}{\partial z} \right) \right\}_{i,j,k}$$

$$\left(\lambda_z ww\right)_{i,j,k} = \left\langle \left\{ \lambda_z w \right\}_{i,j,k}^{\delta w} \middle| w_{i,j,k}, w_{i,j,k+1} \right\rangle \quad \cdots\cdots (3.65)$$

$$\left\{ \gamma_z \nu_e \left(2\frac{\partial w}{\partial z} \right) \right\}_{i,j,k} = \left\{ \gamma_z \right\}_{i,j,k}^{\delta z} \nu_{e.i,j,k} \left[2\frac{w_{i,j,k+1} - w_{i,j,k}}{\Delta z_k} \right] \quad \cdots\cdots (3.66)$$

(4) $$④\, QW_{i,j,k} = \left(-\frac{\gamma_v}{\rho}\frac{\partial p}{\partial z} - \gamma_v D_z w - R_z + \gamma_v S_w - \frac{\gamma_v \rho^* g}{\rho} \right)_{i,j,k}$$

$$\left(\frac{\gamma_v}{\rho}\frac{\partial p}{\partial z}\right)_{i,j,k} = \frac{\lfloor \gamma_v \rfloor_{i,j,k}^{\Delta z}}{\rho}\left[\frac{P_{i,j,k} - P_{i,j,k-1}}{\delta z_k}\right] \quad \text{(3. 67)}$$

$$\left(\gamma_v D_z w\right)_{i,j,k} = \lfloor \gamma_v \rfloor_{i,j,k}^{\Delta z} D_{zi,j,k} w_{i,j,k}^{n+1} \quad \text{(3. 68)}$$

$$\left(R_z\right)_{i,j,k} = \frac{1}{2}\frac{\lfloor C_D \rfloor_{i,j,k}^{\Delta z}}{\delta z_k}\left(1-\gamma_{zi,j,k}\right) v_{i,j,k}\sqrt{\left(\left\{u\right\}_{i,j,k}^{\delta x\ \Delta z}\right)^2 + \left(\left\{v\right\}_{i,j,k}^{\delta y\ \Delta z}\right)^2 + w_{i,j,k}^2} \quad \text{(3. 69)}$$

$$\left(\gamma_v S_w\right)_{i,j,k} = \lfloor \gamma_v \rfloor_{i,j,k}^{\Delta z}\left(w_{i,j,k}\frac{q_k + q_{k-1}}{2} + \frac{\nu}{3}\frac{q_k - q_{k-1}}{\delta z_k}\right) \quad \text{(3. 70)}$$

$$\left(\frac{\gamma_v \rho^* g}{\rho}\right)_{i,j,k} = \left\lfloor \gamma_v \frac{\rho^*}{\rho} \right\rfloor_{i,j,k}^{\Delta z} g \quad \text{(3. 71)}$$

3.3.6. Poisson equation of the potential function

When the outside derivative of the Poisson equation of the potential function shown in equation (3. 18) is discretized, if both sides are multiplied by the volume $\Delta x_i \Delta y_j \Delta z_k$ of the cell to maintain the symmetry of the coefficient matrix of the simultaneous linear equation, it becomes

$$\begin{aligned}&\Delta y_j \Delta z_k\left\{\left(\alpha_x \frac{\partial \phi}{\partial x}\right)_{i+1/2,j,k} - \left(\alpha_x \frac{\partial \phi}{\partial x}\right)_{i-1/2,j,k}\right\} + \Delta x_i \Delta z_k\left\{\left(\alpha_y \frac{\partial \phi}{\partial y}\right)_{i,j+1/2,k} - \left(\alpha_y \frac{\partial \phi}{\partial y}\right)_{i,j-1/2,k}\right\} \\ &+ \Delta x_i \Delta y_j\left\{\left(\alpha_z \frac{\partial \phi}{\partial z}\right)_{i,j,k+1/2} - \left(\alpha_z \frac{\partial \phi}{\partial z}\right)_{i,j,k-1/2}\right\} = \Delta x_i \Delta y_j \Delta z_k \beta_{i,j,k}\end{aligned} \quad \text{(3. 72)}$$

where, for simplicity, we summarize each coefficient as α and β. When the inner derivative is discretized, it becomes

$$\Delta y_j \Delta z_k\left(\alpha_x \frac{\partial \phi}{\partial x}\right)_{i+1/2,j,k} = \frac{\Delta y_j \Delta z_k \alpha_{xi+1/2,j,k}}{\delta x_{i+1}}\left(\phi_{i+1,j,k} - \phi_{i,j,k}\right) \quad \text{(3. 73)}$$

$$\Delta y_j \Delta z_k\left(\alpha_x \frac{\partial \phi}{\partial x}\right)_{i-1/2,j,k} = \frac{\Delta y_j \Delta z_k \alpha_{xi-1/2,j,k}}{\delta x_i}\left(\phi_{i,j,k} - \phi_{i-1,j,k}\right) \quad \text{(3. 74)}$$

$$\Delta x_i \Delta z_k\left(\alpha_y \frac{\partial \phi}{\partial y}\right)_{i,j+1/2,k} = \frac{\Delta x_i \Delta z_k \alpha_{yi,j+1/2,k}}{\delta y_{j+1}}\left(\phi_{i,j+1,k} - \phi_{i,j,k}\right) \quad \text{(3. 75)}$$

$$\Delta x_i \Delta z_k \left(\alpha_y \frac{\partial \phi}{\partial y} \right)_{i,j-1/2,k} = \frac{\Delta x_i \Delta z_k \alpha_{yi,j-1/2,k}}{\delta y_j} \left(\phi_{i,j,k} - \phi_{i,j-1,k} \right) \quad \text{(3. 76)}$$

$$\Delta x_i \Delta y_j \left(\alpha_z \frac{\partial \phi}{\partial z} \right)_{i,j,k+1/2} = \frac{\Delta x_i \Delta y_j \alpha_{zi,j,k+1/2}}{\delta z_{k+1}} \left(\phi_{i,j,k+1} - \phi_{i,j,k} \right) \quad \text{(3. 77)}$$

$$\Delta x_i \Delta y_j \left(\alpha_z \frac{\partial \phi}{\partial z} \right)_{i,j,k-1/2} = \frac{\Delta x_i \Delta y_j \alpha_{zi,j,k-1/2}}{\delta z_k} \left(\phi_{i,j,k} - \phi_{i,j,k-1} \right) \quad \text{(3. 78)}$$

When these equations are integrated and written as simultaneous linear equations, it becomes as follows.

$$A^{KM}_{i,j,k}\phi_{i,j,k-1} + A^{JM}_{i,j,k}\phi_{i,j-1,k} + A^{IM}_{i,j,k}\phi_{i-1,j,k} + A^{D}_{i,j,k}\phi_{i,j,k} + A^{IP}_{i,j,k}\phi_{i+1,j,k} + A^{JP}_{i,j,k}\phi_{i,j+1,k} + A^{KP}_{i,j,k}\phi_{i,j,k+1} = B_{i,j,k} \quad \text{(3. 79)}$$

where

$$A^{KM}_{i,j,k} = \frac{\Delta x_i \Delta y_j \alpha_{zi,j,k-1/2}}{\delta z_k} \quad \text{(3. 80)}$$

$$A^{JM}_{i,j,k} = \frac{\Delta x_i \Delta z_z \alpha_{yi,j-1/2,k}}{\delta y_j} \quad \text{(3. 81)}$$

$$A^{IM}_{i,j,k} = \frac{\Delta y_j \Delta z_z \alpha_{xi-1/2,j,k}}{\delta x_i} \quad \text{(3. 82)}$$

$$A^{IP}_{i,j,k} = \frac{\Delta y_j \Delta z_z \alpha_{xi+1/2,j,k}}{\delta x_{i+1}} \quad \text{(3. 83)}$$

$$A^{JP}_{i,j,k} = \frac{\Delta x_i \Delta z_z \alpha_{yi,j+1/2,k}}{\delta y_{j+1}} \quad \text{(3. 84)}$$

$$A^{KP}_{i,j,k} = \frac{\Delta x_i \Delta y_j \alpha_{zi,j,k+1/2}}{\delta z_{k+1}} \quad \text{(3. 85)}$$

$$A^{D}_{i,j,k} = -\left(A^{KM}_{i,j,k} + A^{JM}_{i,j,k} + A^{IM}_{i,j,k} + A^{IP}_{i,j,k} + A^{JP}_{i,j,k} + A^{KP}_{i,j,k} \right) \quad \text{(3. 86)}$$

$$\alpha_{xi-1/2,j,k} = \gamma_{xi,j,k} \left\lfloor \frac{\gamma_v}{\lambda_v} \right\rfloor^{\Delta x}_{i,j,k} \quad \text{(3. 87)}$$

$$\alpha_{yi,j-1/2,k} = \gamma_{yi,j,k} \left\lfloor \frac{\gamma_v}{\lambda_v} \right\rfloor^{\Delta y}_{i,j,k} \quad \text{(3. 88)}$$

$$\alpha_{zi,j,k-1/2} = \gamma_{zi,j,k} \left\lfloor \frac{\gamma_v}{\lambda_v} \right\rfloor^{\Delta z}_{i,j,k} \quad \text{(3. 89)}$$

$$B_{i,j,k} = \Delta x_i \Delta y_j \Delta z_k \left[\left(\gamma_v S_\rho \right)_{i,j,k} - \left(\begin{array}{l} \dfrac{\gamma_{xi+1,j,k} \tilde{u_{i+1,j,k}} - \gamma_{xi,j,k} \tilde{u_{i,j,k}}}{\Delta x_i} \\ + \dfrac{\gamma_{yi,j+1,k} \tilde{v_{i,j+1,k}} - \gamma_{yi,j,k} \tilde{v_{i,j,k}}}{\Delta y_k} \\ + \dfrac{\gamma_{zi,j,k+1} \tilde{w_{i,j,k+1}} - \gamma_{zi,j,k} \tilde{w_{i,j,k}}}{\Delta z_k} \end{array} \right) \right] \quad \text{..... (3. 90)}$$

3.3.7. Flow velocity and pressure correction equations

The following shows discretization of the flow velocity and pressure correction equations in the equations (3. 19) to (3.22).

$$u_{i,j,k}^{n+1} = \tilde{u_{i,j,k}} + \left[\frac{\gamma_v}{\lambda_v}\right]_{i,j,k}^{\Delta x} \frac{\phi_{i,j,k} - \phi_{i-1,j,k}}{\delta x_i} \quad \text{(3. 91)}$$

$$v_{i,j,k}^{n+1} = \tilde{v_{i,j,k}} + \left[\frac{\gamma_v}{\lambda_v}\right]_{i,j,k}^{\Delta y} \frac{\phi_{i,j,k} - \phi_{i,j-1,k}}{\delta y_j} \quad \text{(3. 92)}$$

$$w_{i,j,k}^{n+1} = \tilde{w_{i,j,k}} + \left[\frac{\gamma_v}{\lambda_v}\right]_{i,j,k}^{\Delta z} \frac{\phi_{i,j,k} - \phi_{i,j,k-1}}{\delta z_k} \quad \text{(3. 93)}$$

$$P_{i,j,k}^{n+1} = P_{i,j,k} - \frac{\rho}{\Delta t}\phi_{i,j,k} \quad \text{(3. 94)}$$

3.3.8. Advection equation of VOF function F

In the VOF method, the behavior of the free surface is analyzed by sequentially calculating the advection equation of the flag NF indicating the direction of the surface and the VOF function F. For this reason, the orientation of the surface and the discretization of the advection equation are very closely related. Hereinafter, a method of determining the orientation of the surface is shown, followed by discretization of the advection equation by a method considering the donor-acceptor method and the inclination of the interface.

(1) Method of determining the direction of the surface

In the VOF method, fluid cell, gas cell and surface cell are classified for each calculation cell, and the surface cell has a discretized orientation (perpendicular to the x axis, perpendicular to the y axis or perpendicular to the z axis). The flag NF is defined as

NF = 0: fluid cell

NF = 1: Surface cell (Surface is perpendicular to x axis and fluid exists in negative direction of x axis)

NF = 2: Surface cell (the surface is perpendicular to the x axis and fluid exists in the positive direction of the x axis)

NF = 3: surface cell (the surface is perpendicular to the y-axis and fluid exists in the y-axis negative direction)

NF = 4: surface cell (the surface is perpendicular to the y axis and fluid exists in the positive direction of the y axis)

NF = 5: Surface cell (surface is perpendicular to z axis and fluid exists in negative direction of z axis)

NF = 6: surface cell (the surface is perpendicular to the z axis and fluid exists in the positive direction of the z axis)

NF = 8: gas cell

Classification of cells by flag NF is shown in Figure 0-3-3(a).

When classifying computation cells into fluid cells, gas cells and surface cells, it is necessary to process boundary conditions on the free surface. As a condition classified as a surface cell, in addition to being 0 <F <1, the surface cell needs to be sandwiched between a fluid cell and a gas cell. Surface cells satisfying these conditions are determined using the following steps.

① All computation cells are fluid cells

② If F = 0, make it a gas cell

③ The cell adjacent to the gas cell shall be the surface cell

④ Cells among the surface cells that are not adjacent to the fluid cell are regarded as gas cells

⑤ Search cells that are not sandwiched between the fluid cell and the gas cell in the surface cell

⑥ Change the cell searched for in ⑤ to a gas cell

⑦ Repeat ③ to ⑤ until there are no more cells to be changed by operation of ⑥

Cells with 0 <F <1 in the example shown in Figure 0-3-3(b) are not included in the surface cell determined by these processes. However, in the example shown in Figure 0-3-3(b), droplets and bubbles that are below the grid resolution are ignored. This is an unavoidable process in order to proceed with the calculation stably.

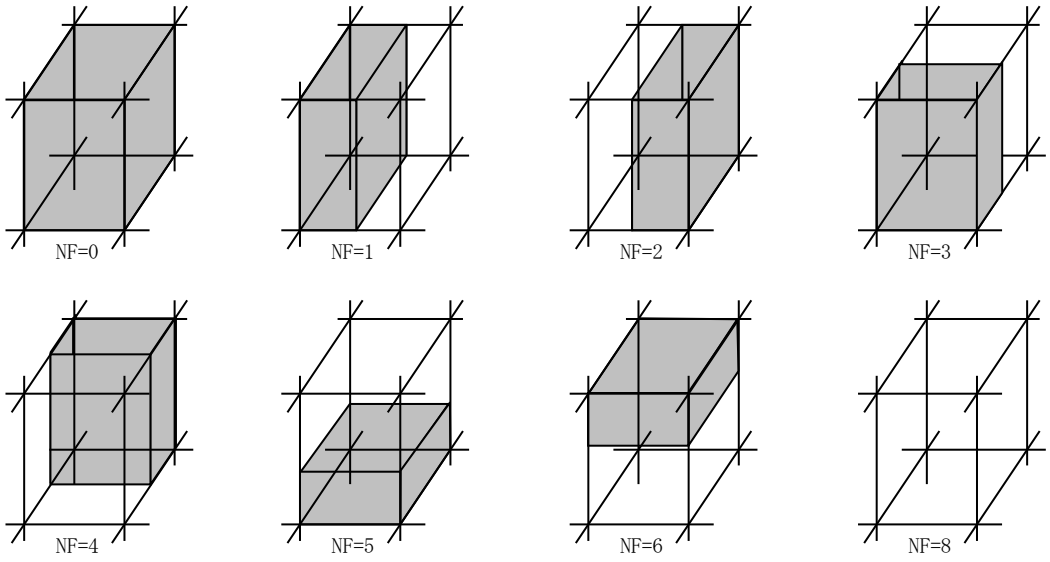
The orientation of the surface of the cell classified as a surface cell in the above processing was determined by the following method.

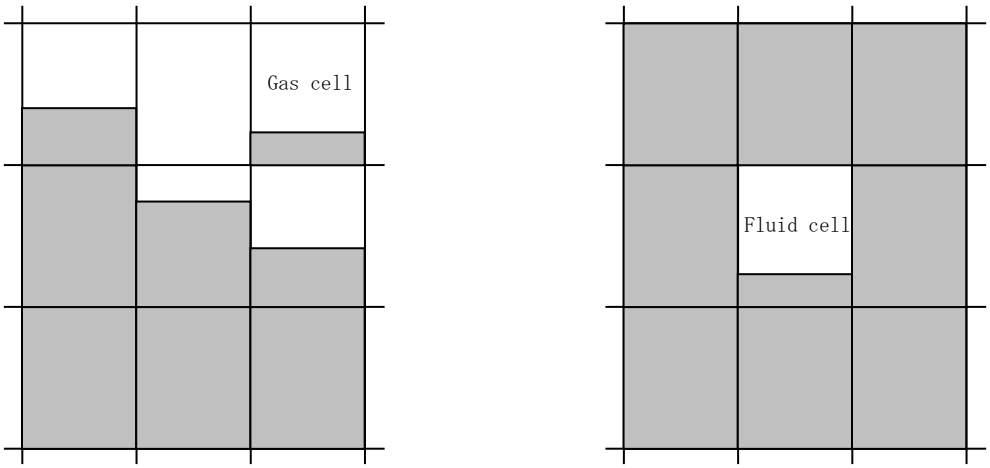
① If the alignment of fluid cell - surface cell - gas cell is unique, it shall be the direction

(2) Fluid cell - Surface cell - Assume that there is a fluid in a direction in which the value of the surrounding F is large in the direction satisfying the gas cell

① If the alignment of fluid cell - surface cell - gas cell is unique, it shall be the direction

② If Fluid Cell - Surface Cell - Gas Cell is satisfied, assume that there is a fluid in the direction where the value of the surrounding F is large.





(a) Classification of cells by flag NF (b) An example where the surface cell does not become a surface cell with 0. 0 <F <1.0.

Figure 0-3-3 Classification of calculation cells

(2) Donor acceptor method

In the donor-acceptor method, the apparent windward and leeward differences are selectively used depending on the direction of the surface, and the surface is prevented from being blurred. The advection amount is determined with the amount of water contained in the donor cell as the upper limit. The concept of the donor-acceptor method is shown in Figure 0-3-4, and the discrete equation is shown below.

The amount of flowing in and out through the interface is denoted by the symbol in Figure 0-3-2, and the amount to be generated and extinguished in the control volume is written as $QF_{i,j,k}$, and the the equation (3.37) is discretized as follows.

$$F_{i,j,k}^{n+1} = F_{i,j,k}^{n} + \frac{1}{\gamma_{v.i,j,k}} \left(\begin{array}{l} \frac{1}{\Delta x_i}\left[FLFU_{i+1,j,k} - FLFU_{i,j,k}\right] + \frac{1}{\Delta y_j}\left[FLFV_{i,j+1,k} - FLFV_{i,j,k}\right] \\ + \frac{1}{\Delta z_k}\left[FLFW_{i,j,k+1} - FLFW_{i,j,k}\right] + \Delta t QF_{i,j,k} \end{array} \right) \quad \text{..... (3. 95)}$$

Since the donor-acceptor method is used, Δt is included in the inlet / outlet amount from the interface. Only the terms $FLFU_{i,j,k}$ and $\Delta tQF_{i,j,k}$ B are shown below. Since F is defined at the center of the

cell, there is no direction dependence in the discrete equation of the amount of inlet and outlet through the interface, and it is possible to describe by changing the suffix.

① $FLFU_{i,j,k} = \left(-\Delta t \gamma_x uF\right)_{i,j,k}$

$$\left(\Delta t \gamma_x uF\right)_{i,j,k} = sign(C)\min(F_{AD}|C| + CFX, F_D \Delta x_D \gamma_{vD}) \quad \text{(3. 96)}$$

$$CFX = \max\left[(F_{DM} - F_{AD})|C| - (F_{DM} - F_D)\Delta x_D \gamma_{vD}, 0\right]$$

$$C = \Delta t \gamma_{xi,j,k} u_{i,j,k}$$

$$F_{AD} = \begin{cases} F_D: \text{The interface is located perpendicular to the water surface on the donor side} \\ F_A: \text{Other than those above} \end{cases}$$

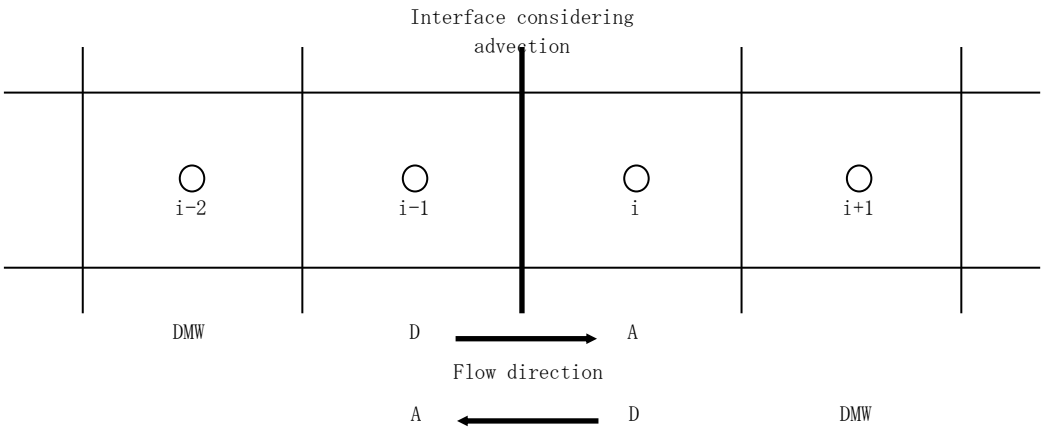
$$F_{DM} = \max(F_{DMW}, F_D)$$

$C \geq 0$: $DMW = i-2$, $D = i-1$, $A = i$

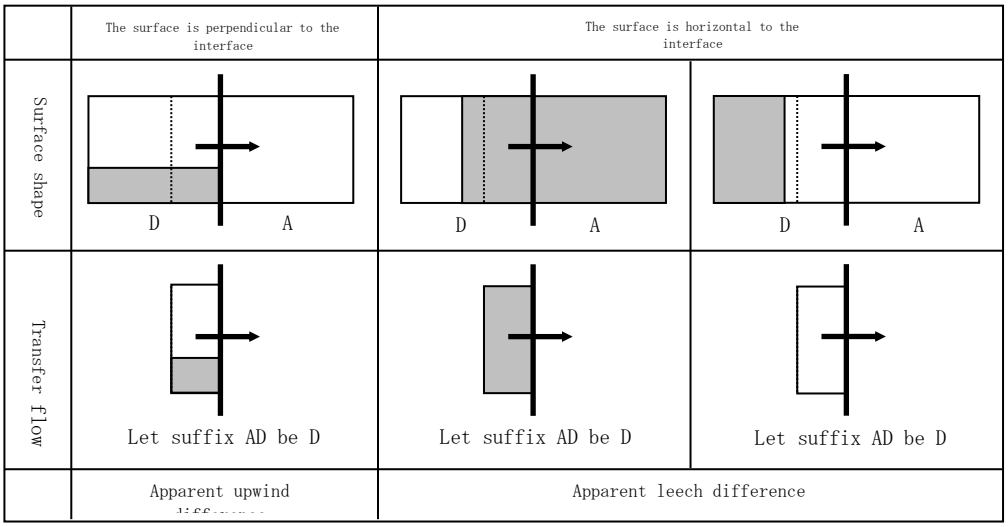
$C < 0$: $DMW = i+1$, $D = i$, $A = i-1$

② $QF_{i,j,k} = \left(S_F\right)_{i,j,k}$

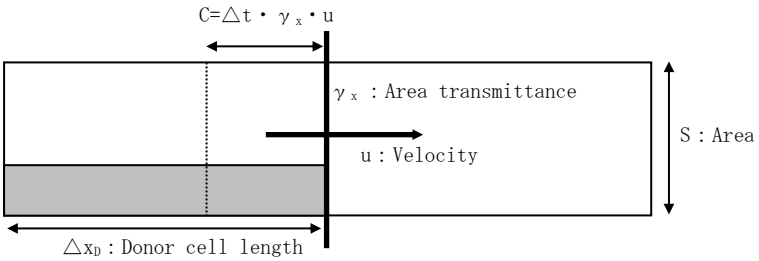
$$\left(S_F\right)_{i,j,k} = F_{i,j,k} q_{i,j,k} \quad \text{(3. 97)}$$



(a) Changes in suffix due to positive and negative flow velocities at the interface



(b) Classification of surface cells based on interfaces and advection



Transfer flow rate＝min (F_{AD}・C・S , F_D・$\triangle x_D$・γ_{vD}・S)

Amount of water to be advected Maximum amount of water that can be

(c) Restricted advection (Suffix AD is A or D)

Figure 0-3-4 Concept of donor-acceptor method

(3) Method considering inclination of interface

In the donor-acceptor method, as shown in Figure 0-3-5(b), since the interface is set perpendicular to the calculation grid, there is a problem that the accuracy of the flow rate decreases. In order to deal with this problem, as shown in Figure 0-3-5(c), a method 14) for introducing the interface into a linear function (plane in three dimensions) and considering its inclination was introduced. The inclination is taken into account for the advection term of the VOF function F. Calculation on the flow rate at the interface and pressure is performed by the existing method.

The calculation procedure of the method considering the inclination of the interface is as follows.

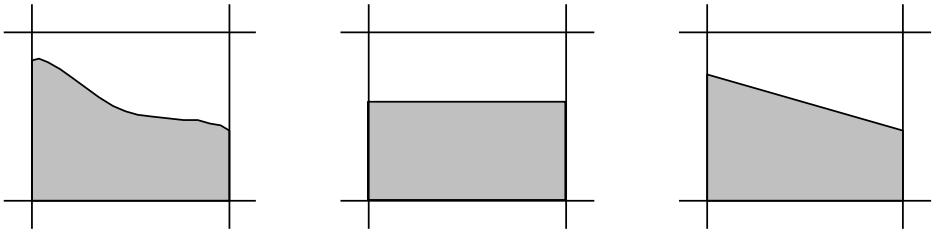
① Calculation of normal vector: Calculate the normal vector of the interface by numerically differentiating the VOF function F

② Classification of interface pattern: Classify interface pattern from normal direction and advection direction

③ Determination of interface: The interface equation is determined from the amount of fluid contained in the calculation cell

④ Calculation of advection amount: Calculate the advection amount from interface equation and advection velocity

For the sake of simplicity, it is described below in two dimensions (interface is a linear function).



(a) Before modeling (b) Donor · Acceptor method (c) Method considering inclination of interface

Figure 0-3-5 Modeling the interface

①Calculation of normal vector

As a method of calculating the normal vector of the interface, as shown in Figure III-3-6

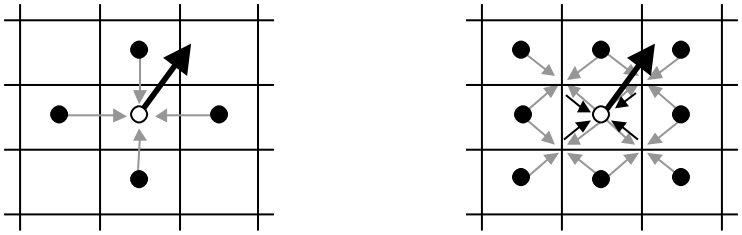
Calculation at cell center

Calculation at grid points, and value at cell center shall be the average of values of surrounding grid points

could be considerd, but from the consistency with the secured position of the boundary value of CADMAS-SURF/3D-MG, it was calculated at the cell center using the following equation.

$$n_{xi,k} = -\frac{\partial F}{\partial x} = \frac{F_{i-1,k} - F_{i+1,k}}{\delta x_i + \delta x_{i+1}} \quad \text{(3. 98)}$$

$$n_{zi,k} = -\frac{\partial F}{\partial z} = \frac{F_{i,k-1} - F_{i,k+1}}{\delta z_k + \delta z_{k+1}} \quad \text{(3. 99)}$$



(a)Cell centererd (b) Grid points

Figure 0-3-6 Calculation of normal vector

②Classification of interface pattern

In order to avoid complicated processing, classify the interface pattern from the normal direction and the advection direction. As shown in Figure 0-3-7, A flag ISW indicating which side of the cell on the side supplying the fluid flows out from the left or right side is set as the advection direction as the horizontal direction. The sign of the flag ISW and the normal vector component are reversed from the direction of the normal vector so that the component of the normal vector at the interface is always positive (See Figure 0-3-8).

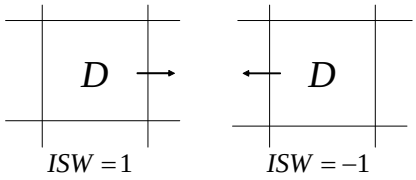


Figure 0-3-7 An interface to be considered as a target cell

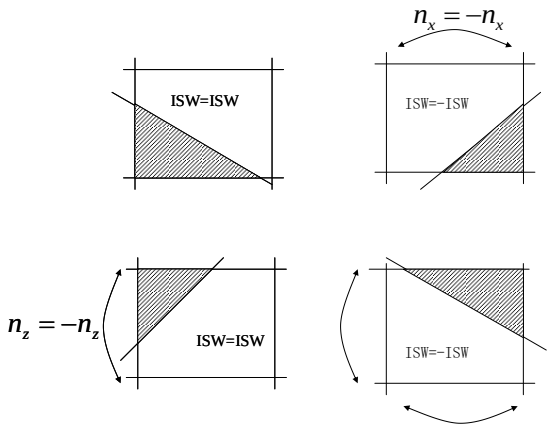
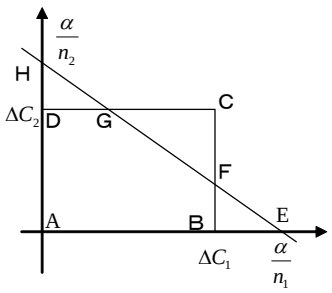


Figure 0-3-8 Positive and negative reversal

By the above processing, when the horizontal coordinate is c_1, the vertical coordinate is c_2, and the grid interval in each direction is Δc_1 and Δc_2, the interface (linear function) to be considered as the target cell is as shown in the figure below .



$$n_1 c_1 + n_2 c_2 = \alpha \geq 0$$

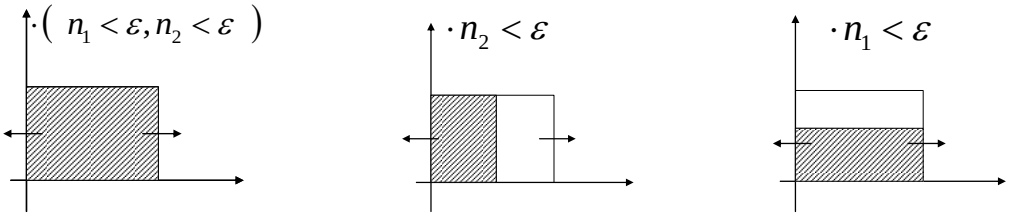
$$c_2 = -\frac{n_1}{n_2} c_1 + \frac{\alpha}{n_2}$$

$$c_1 = -\frac{n_2}{n_1} c_2 + \frac{\alpha}{n_1}$$

Figure 0-3-9 An interface to be considered as a target cell

If the origin is zero and the area occupied by the fluid (ABFGD) is greater than or equal to zero, the constant α of the linear function in Figure 0-3-8 is greater than or equal to zero.

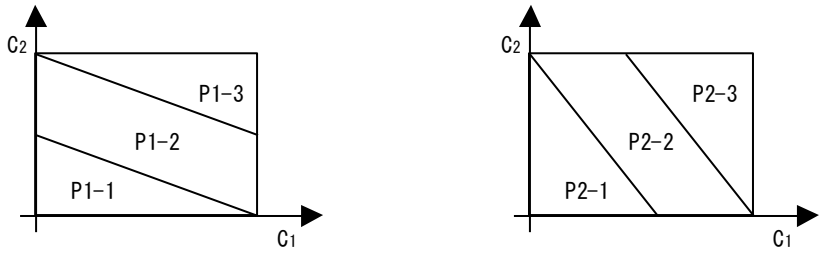
Since the interface to be considered as the target cell has been determined, the interface pattern is finally classified.If the length of the normal vector is zero (fluid cell or gas cell) or vertical or horizontal to the advection direction, calculate it in the same way as the donor acceptor method (see Figure 0-3-10).



(a) If the length of the normal vector is zero (b) Vertical to the advection direction (c) Horizontal in the advection direction

Figure 0-3-10 A pattern to be calculated in the same way as the donor-acceptor method

When the interface is inclined, it is classified into the following six patterns.



(a) $n_1\Delta C_1 \le n_2\Delta C_2$ (b) $n_1\Delta C_1 > n_2\Delta C_2$

Figure 0-3-11 Pattern when the interface is inclined

Assuming that the area occupied by the fluid is V_{give}, the details of this classification are as follows.

●Figure 0-3-11(a) （$n_1\Delta C_1 \le n_2\Delta C_2$）

・P1-1 : Assuming that the area when the interface passes through point B is $V_{\lim}$, by $\frac{\alpha}{n_1} = \Delta C_1$,

$V_{\lim} = \frac{\alpha}{n_1} \times \frac{\alpha}{n_2} \times \frac{1}{2} = \frac{1}{2}\frac{\alpha^2}{n_1 n_2}$. Therefore, if it is $V_{give} \le V_{\lim}$, this is a pattern of P1-1.

Values required later are

$H_1 = H_2 = 0.0$

where

$H_1 = H(\alpha - n_1\Delta c_1)$, $H_2 = H(\alpha - n_2\Delta c_2)$, $H(x) = 0$ *for* $x \le 0$, 1 *for* $x > 0$.

・P1-1 : Assuming that the area when the interface passes through point D is $V_{\lim}$, by $\frac{\alpha}{n_2} = \Delta C_2$,

to $V_{\lim} = \frac{\Delta c_1}{2}\left(2\Delta c_2 - \frac{n_1}{n_2}\Delta c_1\right)$. Therefore, if it is $V_{give} > V_{\lim}$, this is a pattern of P1-3.

Values required later are $H_1 = 1.0$, $H_2 = 1.0$ ..

・ P1-2: If it is not a pattern of P1-1 and P1-3, it is P1-2.

Values required later are $H_1 = 1.0$, $H_2 = 0.0$ ..

●Figure 0-3-11(b) （$n_1\Delta C_1 > n_2\Delta C_2$）

・P2-1 : Assuming that the area when the interface passes through point D is $V_{\lim}$, by $\frac{\alpha}{n_2} = \Delta C_2$,

$V_{\lim} = \frac{\alpha}{n_1} \times \frac{\alpha}{n_2} \times \frac{1}{2} = \frac{1}{2} \frac{\alpha^2}{n_1 n_2}$. Therefore, if it is $V_{give} \le V_{\lim}$, this is a pattern of P2-1.

Values required later are $H_1 = H_2 = 0.0$ ..

・P2-3：Assuming that the area when the interface passes through point B is $V_{\lim}$, by $\frac{\alpha}{n_1} = \Delta C_1$,

$V_{\lim} = \frac{\Delta c_2}{2} \left(2\Delta c_1 - \frac{n_2}{n_1} \Delta c_2 \right)$. Therefore, if it is $V_{give} > V_{\lim}$, this is a pattern of P2-3.

Values required later are $H_1 = 1.0$, $H_2 = 1.0$ ..

・ P2-2: If it is not a pattern of P2-1 and P2-3, it is P2-2.

Values required later are $H_1 = 0.0$, $H_2 = 1.0$ ..

③Determination of interface

From a quantity of fluid contained in the calculation cell, α of the interface equation

$$n_1 c_1 + n_2 c_2 = \alpha \quad \text{..... (3. 100)}$$

is determined.

The amount $V(\alpha)$ of the fluid is calculated using the aforementioned H_1 and H_2 is

$$\begin{aligned} V(\alpha) &= \Delta AEH - H_1 \times \Delta BEF - H_2 \times \Delta DGH \\ &= \frac{1}{2} \times \frac{\alpha}{n_1} \times \frac{\alpha}{n_2} - H_1 \times \frac{1}{2} \left(\frac{\alpha}{n_1} - \Delta C_1 \right) \left(-\frac{n_1}{n_2} \Delta C_1 + \frac{\alpha}{n_2} \right) \\ &\qquad - H_2 \times \frac{1}{2} \left(\frac{\alpha}{n_2} - \Delta C_2 \right) \left(-\frac{n_2}{n_1} \Delta C_2 + \frac{\alpha}{n_1} \right) \\ &= \frac{\alpha^2}{2 n_1 \eta_2} - H_1 \times \frac{1}{2} \left(\frac{\alpha - n_1 \Delta C_1}{n_1} \right) \left(\frac{\alpha - n_1 \Delta C_1}{n_2} \right) - H_2 \times \frac{1}{2} \left(\frac{\alpha - n_2 \Delta C_2}{n_2} \right) \left(\frac{\alpha - n_2 \Delta C_2}{n_1} \right) \\ &= \frac{\alpha^2}{2 n_1 n_2} - H_1 \times \frac{1}{2 n_1 n_2} (\alpha - n_1 \Delta C_1)^2 - H_2 \times \frac{1}{2 n_1 n_2} (\alpha - n_2 \Delta C_2)^2 \\ &= \frac{1}{2 n_1 n_2} \left\{ \alpha^2 - H_1 (\alpha - n_1 \Delta C_1)^2 - H_2 (\alpha - n_2 \Delta C_2)^2 \right\} \end{aligned} \quad \text{.. (3. 101)}$$

where α can be determined by solving a quadratic equation

$$V_{give} = V(\alpha) = A\alpha^2 + B\alpha + C \quad \text{..... (3. 102)}$$

$$\begin{cases} A = (1 - H_1 - H_2) / 2 n_1 n_2 \\ B = (2 H_1 n_1 \Delta c_1 + 2 H_2 n_2 \Delta c_2) / 2 n_1 n_2 \\ C = \left(-H_1 {n_1}^2 {\Delta c_1}^2 - H_2 {n_2}^2 {\Delta c_2}^2 \right) / 2 n_1 n_2 \end{cases}$$

④Calculation of advection amount

Since α has been determined, the coordinate value c_1 in the horizontal direction is taken as

$$\begin{cases} ISW = \ 1 : c_1 = \Delta c_1 - \Delta t \gamma_1 |u| \\ ISW = -1 : c_1 = \Delta t \gamma_1 |u| \end{cases} \quad \text{(3. 103)}$$

and the transfer flow rate V_{flx} is calculated as follows.

●P1-1 or P2-1 (see Figure 0-3-12(a))

$$\begin{cases} c_1 \geq P = \dfrac{\alpha}{n_1} : V_* = 0 \\ c_1 < P = \dfrac{\alpha}{n_1} : V_* = \dfrac{1}{2}(p - c_1)\left(-\dfrac{n_1}{n_2} c_1 + \dfrac{\alpha}{n_2} \right) \end{cases} \quad \text{(3. 104)}$$

$$\begin{cases} ISW = \ 1 : V_{flx} = V_* \\ ISW = -1 : V_{flx} = V_{give} - V_* \end{cases} \quad \text{(3. 105)}$$

●P1-3 or P2-3 (see Figure 0-3-12(b))

$$\begin{cases} c_1 \leq P = -\dfrac{n_2}{n_1} \Delta c_2 + \dfrac{\alpha}{n_1} : V_* = 0 \\ c_1 > P = -\dfrac{n_2}{n_1} \Delta c_2 + \dfrac{\alpha}{n_1} : V_* = \dfrac{1}{2}(c_1 - p)\left[\Delta c_2 - \left(-\dfrac{n_1}{n_2} c_1 + \dfrac{\alpha}{n_2} \right) \right] \\ V_{**} = c_1 \Delta c_2 - V_* \end{cases} \quad \text{(3. 106)}$$

$$\begin{cases} ISW = \ 1 : V_{flx} = V_{give} - V_{**} \\ ISW = -1 : V_{flx} = V_{**} \end{cases} \quad \text{(3. 107)}$$

●P1-2 (see Figure 0-3-12(c))

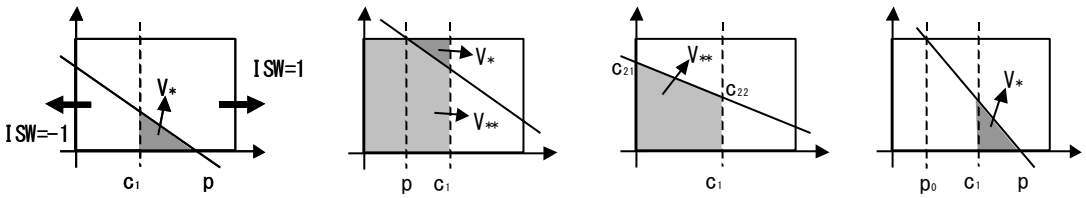
$$\begin{cases} c_{21} = -\dfrac{n_1}{n_2} 0 + \dfrac{\alpha}{n_2} = \dfrac{\alpha}{n_2} \\ c_{22} = -\dfrac{n_1}{n_2} c_1 + \dfrac{\alpha}{n_2} \\ V_{**} = \dfrac{1}{2} c_1 (c_{21} + c_{22}) \end{cases} \quad \text{(3. 108)}$$

$$\begin{cases} ISW = \ 1 : V_{flx} = V_{give} - V_{**} \\ ISW = -1 : V_{flx} = V_{**} \end{cases} \quad \text{(3. 109)}$$

●P2- 2 (see Figure 0-3-12(d))

$$\begin{cases} p_0 < c_1 \,\&\, c_1 \geq P = \dfrac{\alpha}{n_1} : V_* = 0 \\ p_0 < c_1 \,\&\, c_1 < P = \dfrac{\alpha}{n_1} : V_* = \dfrac{1}{2}\left(p - c_1\right)\left(-\dfrac{n_1}{n_2} c_1 + \dfrac{\alpha}{n_2} \right) \\ p_0 \geq c_1 \qquad\qquad : V_* = V_{give} - p_0 \Delta c_2 \end{cases} \quad \text{(3. 110)}$$

$$\begin{cases} ISW = \ \ 1 : V_{flx} = V_* \\ ISW = -1 : V_{flx} = V_{give} - V_* \end{cases} \quad \text{(3. 111)}$$



(a) P1-1 or P2-1 (b) P1-3 or P2-3 (c) P1-2 (d) P2-2

Figure 0-3-12 Calculation of advection amount

3.3.9. k - ε 2 equation Model

The amount of inlet and outlet through the interface is described by $FLKU_{i,j,k}$ and $FLEU_{i,j,k}$ like the symbols in Figure 0-3-2, and the amount to be generated and extinguished in the control volume is written as $QK_{i,j,k}$ and $QE_{i,j,k}$, and the equations (2. 22) and (2.23) are discretized follows.

$$k_{i,j,k}^{n+1} = k_{i,j,k}^{n} + \frac{\Delta t}{\gamma_{vi,j,k}} \left(\begin{array}{l} \dfrac{1}{\Delta x_i}\left[FLKU_{i+1,j,k} - FLKU_{i,j,k} \right] + \dfrac{1}{\Delta y_j}\left[FLKV_{i,j+1,k} - FLKV_{i,j,k} \right] \\ + \dfrac{1}{\Delta z_k}\left[FLKW_{i,j,k+1} - FLKW_{i,j,k} \right] + QK_{i,j,k} \end{array} \right) \quad \text{(3. 112)}$$

$$\varepsilon_{i,j,k}^{n+1} = \varepsilon_{i,j,k}^{n} + \frac{\Delta t}{\gamma_{vi,j,k}} \left(\begin{array}{l} \frac{1}{\Delta x_i}\left[FLEU_{i+1,j,k} - FLEU_{i,j,k}\right] + \frac{1}{\Delta y_i}\left[FLEV_{i,j+1,k} - FLEV_{i,j,k}\right] \\ + \frac{1}{\Delta z_k}\left[FLEW_{i,j,k+1} - FLEW_{i,j,k}\right] + QE_{i,j,k} \end{array} \right) \quad \text{..... (3. 113)}$$

Since k and ε are defined at the center of the cell, there is no direction dependence in the discrete equation of the amount of inlet and outlet through the interface, and it is possible to describe by changing the suffix.

Therefore, only $FLKU_{i,j,k}$, $QK_{i,j,k}$ and $QE_{i,j,k}$ are shown below.

① $FLKU_{i,j,k} = \left(-\gamma_x uk + \gamma_x \nu_k \left[\frac{\partial k}{\partial x} \right] \right)_{i,j,k}$

$$\left(\gamma_x uk\right)_{i,j,k} = \left\langle \left(\gamma_x u\right)_{i,j,k} \middle| k_{i-1,j,k}, k_{i,j,k} \right\rangle \quad \text{..... (3. 114)}$$

$$\left(\gamma_x \nu_k \left[\frac{\partial k}{\partial x} \right] \right)_{i,j,k} = \gamma_{xi,j,k} \left\{ \nu_k \right\}_{i,j,k}^{\Delta x} \left[\frac{k_{i,j,k} - k_{i-1,j,k}}{\delta x_i} \right] \quad \text{..... (3. 115)}$$

② $QK_{i,j,k} = (\gamma_v G_s - \gamma_v \varepsilon)_{i,j,k} = \gamma_{vi,j,k} G_{si,j,k} - \gamma_{vi,j,k} \varepsilon_{i,j,k}$ ..... (3. 116)

③ $QE_{i,j,k} = (\gamma_v C_1 \frac{\varepsilon}{k} G_s - \gamma_v C_2 \frac{\varepsilon^2}{k})_{i,j,k} = \gamma_{vi,j,k} C_1 \frac{\varepsilon_{i,j,k}}{k_{i,j,k}} G_{si,j,k} - \gamma_{vi,j,k} C_2 \frac{\varepsilon_{i,j,k}^2}{k_{i,j,k}}$ ..... (3. 117)

Note that A in the equations (3.116) and (3.117) is $G_{si,k}$ as follows.

$$G_{si,j,k} = \left(\nu_t \left[2\left\{ \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial z}\right)^2 \right\} + \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right)^2 \right] \right)_{i,j,k}$$

$$= \nu_{ti,j,k} \left(\begin{array}{l} 2\left[\left(\dfrac{u_{i+1,j,k} - u_{i,j,k}}{\Delta x_i} \right)^2 + \left(\dfrac{v_{i+1,j,k} - v_{i,j,k}}{\Delta y_j} \right)^2 + \left(\dfrac{w_{i,j,k+1} - w_{i,j,k}}{\Delta z_k} \right)^2 \right] \\ + \left(\dfrac{\left\{ \left\{ u \right\}_{i,j+1,k}^{\Delta y} \right\}_{i,j,k}^{\delta x} - \left\{ \left\{ u \right\}_{i,j,k}^{\Delta y} \right\}_{i,j,k}^{\delta x}}{\Delta y_j} + \dfrac{\left\{ \left\{ v \right\}_{i+1,j,k}^{\Delta x} \right\}_{i,j,k}^{\delta y} - \left\{ \left\{ v \right\}_{i,j,k}^{\Delta x} \right\}_{i,j,k}^{\delta y}}{\Delta x_i} \right)^2 \\ + \left(\dfrac{\left\{ \left\{ v \right\}_{i,j,k+1}^{\Delta z} \right\}_{i,j,k}^{\delta y} - \left\{ \left\{ v \right\}_{i,j,k}^{\Delta z} \right\}_{i,j,k}^{\delta y}}{\Delta z_k} + \dfrac{\left\{ \left\{ w \right\}_{i,j+1,k}^{\Delta y} \right\}_{i,j,k}^{\delta z} - \left\{ \left\{ w \right\}_{i,j,k}^{\Delta y} \right\}_{i,j,k}^{\delta z}}{\Delta y_j} \right)^2 \\ + \left(\dfrac{\left\{ \left\{ w \right\}_{i+1,j,k}^{\Delta x} \right\}_{i,j,k}^{\delta z} - \left\{ \left\{ w \right\}_{i,j,k}^{\Delta x} \right\}_{i,j,k}^{\delta z}}{\Delta x_i} + \dfrac{\left\{ \left\{ u \right\}_{i,j,k+1}^{\Delta z} \right\}_{i,j,k}^{\delta x} - \left\{ \left\{ u \right\}_{i,j,k}^{\Delta z} \right\}_{i,j,k}^{\delta x}}{\Delta z_k} \right)^2 \end{array} \right) \quad \cdots\cdots\cdots\cdots \quad (3.\ 118)$$

3.3.10. Energy equation

The amount of inlet and outlet through the interface is described by $FLTU_{i,j,k}$ and $FLTU_{i,j,k}$ like the symbols in Figure 0-3-2, and the amount to be generated and extinguished in the control volume is written as $QT_{i,j,k}$, and the equations (2. 32) is discretized follows.

$$T_{i,j,k}^{n+1} = T_{i,j,k}^{n} + \frac{\Delta t}{\gamma_{vi,j,k}} \left(\begin{array}{l} \dfrac{1}{\Delta x_i}\left[FLTU_{i+1,j,k} - FLTU_{i,j,k} \right] + \dfrac{1}{\Delta y_j}\left[FLTV_{i,j+1,k} - FLTV_{i,j,k} \right] \\ + \dfrac{1}{\Delta z_k}\left[FLTW_{i,j,k+1} - FLTW_{i,j,k} \right] + QT_{i,j,k} \end{array} \right) \quad \cdots\cdots\cdots \quad (3.\ 119)$$

Since T is defined at the center of the cell, there is no direction dependence in the discrete equation of the amount of inlet and outlet through the interface, and it is possible to describe by changing the suffix.

Therefore, only $FLTU_{i,j,k}$ and $QT_{i,j,k}$ are shown below.

① $FLTU_{i,j,k} = \left(-\gamma_x uT + \dfrac{\gamma_x \lambda_e}{\rho c_p}\left[\dfrac{\partial T}{\partial x} \right] \right)_{i,j,k}$

$$\left(\gamma_x uT \right)_{i,j,k} = \left\langle \left(\gamma_x u \right)_{i,j,k} \middle| T_{i-1,j,k}, T_{i,j,k} \right\rangle \quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (3.\ 120)$$

$$\left(\frac{\gamma_x\lambda_e}{\rho c_p}\left[\frac{\partial T}{\partial x}\right]\right)_{i,j,k}=\frac{\gamma_{xi,j,k}\{\lambda_e\}^{\Delta x}_{i,j,k}}{\rho c_p}\left[\frac{T_{i,j,k}-T_{i-1,j,k}}{\delta x_i}\right] \quad \text{..... (3. 121)}$$

$$\text{② } QT_{i,j,k}=(\rho c_p Tq(z,t))_{i,j,k}=\rho c_p T_{i,j,k}q(z,t)_{i,j,k} \quad \text{..... (3. 122)}$$

3.3.11. Concentration transport equations

The amount of inlet and outlet through the interface is described by $FLCU_{i,j,k}$ and $FLCU_{i,j,k}$ like the symbols in Figure 0-3-2, and the amount to be generated and extinguished in the control volume is written as $QC_{i,j,k}$, and the equations (2. 35) is discretized follows.

$$c^{n+1}_{i,j,k}=c^{n}_{i,j,k}+\frac{\Delta t}{\gamma_{vi,j,k}}\left(\begin{array}{l}\frac{1}{\Delta x_i}\left[FLCU_{i+1,j,k}-FLCU_{i,j,k}\right]+\frac{1}{\Delta y_j}\left[FLCV_{i,j+1,k}-FLCV_{i,j,k}\right]\\+\frac{1}{\Delta z_k}\left[FLCW_{i,j,k+1}-FLCW_{i,j,k}\right]+QC_{i,j,k}\end{array}\right) \quad \text{..... (3. 123)}$$

The suffix indicating the concentration component is omitted. Since C is defined at the center of the cell, there is no direction dependence in the discrete equation of the amount of inlet and outlet through the interface, and it is possible to describe by changing the suffix.

Therefore, only $FLCU_{i,j,k}$ and $QC_{i,j,k}$ are shown below.

$$\text{① } FLCU_{i,j,k}=\left(-\gamma_x uc+\gamma_x D_e\left[\frac{\partial c}{\partial x}\right]\right)_{i,j,k}$$

$$(\gamma_x uc)_{i,j,k}=\left\langle(\gamma_x u)_{i,j,k}\middle|c_{i-1,j,k},c_{i,j,k}\right\rangle \quad \text{..... (3. 124)}$$

$$\left(\gamma_x D_e\left[\frac{\partial c}{\partial x}\right]\right)_{i,j,k}=\gamma_{xi,j,k}\{D_e\}^{\Delta x}_{i,j,k}\left[\frac{c_{i,j,k}-c_{i-1,j,k}}{\delta x_i}\right] \quad \text{..... (3. 125)}$$

$$\text{② } QC_{i,j,k}=(cq(z,t))_{i,j,k}=c_{i,j,k}q(z,t)_{i,j,k} \quad \text{..... (3. 126)}$$

3.4. Solving simultaneous linear equations

In the SMAC method, the solution of asymmetric simultaneous linear equations on the potential function ϕ occupies much of the total computation time (70% to over 90%), and preservation of scalar quantity largely depends on the accuracy of continuous equations. Especially in the VOF method, maintaining the accuracy of the conservation law of the VOF function F value is important because it is agreed to keep the total water volume of the calculation system.

In CADMAS-SURF/3D-MG, BCGSTAB method with preprocessing with excellent convergence and stability was adopted as a solution of asymmetric simultaneous linear equation [15)]

The algorithm of the BCGSTAB method with preprocessing is shown below.

$\mathbf{x}_0 =$ Initial solution vector, $\mathbf{r}_0 = \mathbf{p}_0 = M^{-1}(\mathbf{b} - A\mathbf{x}_0)$, $k = 0$

① $\alpha_k = \dfrac{(\mathbf{r}_0, \mathbf{r}_k)}{(\mathbf{r}_0, M^{-1}A\mathbf{p}_k)}$

② $\mathbf{e}_k = \mathbf{r}_k - \alpha_k M^{-1}A\mathbf{p}_k$

③ $\omega_k = \dfrac{(\mathbf{e}_k, M^{-1}A\mathbf{e}_k)}{(M^{-1}A\mathbf{e}_k, M^{-1}A\mathbf{e}_k)}$

④ $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k + \omega_k \mathbf{e}_k$

⑤ $\mathbf{r}_{k+1} = \mathbf{e}_k - \omega_k M^{-1}A\mathbf{e}_k$

⑥ If $(\mathbf{r}_{k+1}, \mathbf{r}_{k+1}) < \varepsilon(\mathbf{b}, \mathbf{b})$, solution finished.

⑦ $\beta_k = \dfrac{(\mathbf{r}_0, \mathbf{r}_{k+1})}{\omega_k(\mathbf{r}_0, M^{-1}A\mathbf{p}_k)}$

⑧ $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k(\mathbf{p}_k - \omega_k M^{-1}A\mathbf{p}_k)$

⑨ $k = k + 1$

⑩ Back to ①

where A is a coefficient matrix of simultaneous linear equations, $\mathbf{x}$ is solution vector, $\mathbf{b}$ is right side of simultaneous linear equation and M is pre-processing matrix..

3.5. Time step width

From the problem of stability in numerical calculation, the following condition is imposed on the time step width Δt.

① CFL condition

$$\Delta t \le \min\left(\frac{\Delta x}{u}, \frac{\Delta y}{v}, \frac{\Delta z}{w}\right) = \Delta t_c \quad \text{(3. 127)}$$

② Stability condition of viscosity term

$$\Delta t \le \frac{1}{2} \frac{1}{\nu\left[\left(\frac{1}{\Delta x}\right)^2 + \left(\frac{1}{\Delta y}\right)^2 + \left(\frac{1}{\Delta z}\right)^2\right]} = \Delta t_\nu \quad \text{(3. 128)}$$

Since it is difficult for the user to determine the time step width Δ t which always satisfies this condition, the function of "automatic time step size" is added in CADMAS-SURF/3D-MG. In the "automatic time step width", the time step width Δt is determined as follows using the safety factor α (0. 0 <α <1.0) input by the user.

$$\Delta t = \alpha \min\left(\Delta t_c, \Delta t_\nu\right) \quad \text{(3. 129)}$$

3.6. Air pressure calculation model

When calculation is performed with respect to a place where air is taken in using CADMAS-SURF/3D-MG, bubbles collapse as it is, numerical noise is generated, and there arises a problem that the wave pressure can not be calculated accurately. For the purpose of solving this problem, we added a function to avoid bubble collapse without resistance by calculating the internal pressure and volume change of the entrapped air and setting the internal pressure at that time as the pressure boundary condition of the fluid surface. Assuming that the entrapped gas changes in the adiabatic state, there is the following relationship between the pressure p of the gas and the volume V.

$$pV^\gamma = \text{Constant value} \quad \text{(3. 130)}$$

where, γ is the specific heat ratio, which is approximately 1. 403 in the case of air.

Therefore, by calculating the constant value of the right side from the initial pressure and volume of the gas and calculating the volume V of the gas at each time step, the pressure p of the gas is obtained.

However, for example, if one continuous gas region divides into two different gas regions at the next time step, the method of determining this constant value in each region becomes a problem. Conversely, the same problem occurs when two different gas regions merge and become one. In order to avoid this problem, we use the following form which differentiates both sides of the above equation with time.

Differentiating both sides with t yields

$$\gamma pV^{\gamma-1}\frac{dV}{dt}+V^{\gamma}\frac{dp}{dt}=0 \quad \text{(3. 131)}$$

and transforming the expression becomes

$$\frac{dp}{dt}=-\gamma\frac{p}{V}\frac{dV}{dt} \quad \text{(3. 132)}$$

where the volume change of the gas can be regarded as the area divided of the movement of the surface of the bubble and decribed as

$$\frac{dV}{dt}=\int_{S} v\cdot dS \quad \text{(3. 133)}$$

Applying Gauss' divergence theorem to this equation, in the discretization, it can be transformed to

$$\frac{dV}{dt}=\int_{V}\nabla\cdot v\,dV=\sum_{ijk\in V}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\Delta V_{ijk} \quad \text{(3. 134)}$$

where ijk is the index of the computation cell and its conceptual diagram is shown in Figure 0-3-13.

Regarding the gas volume V, by describing

$$V=\sum_{ijk\in V}\Delta V_{ijk} \quad \text{(3. 135)}$$

, expressions (3. 134) and (3. 135) are substituted into equation (3. 132) and discretized by Euler

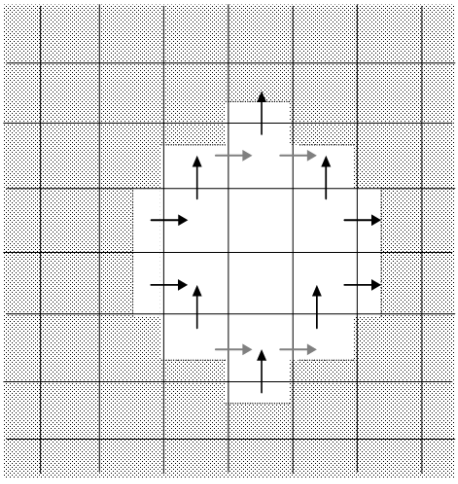
implicit solution as follows.

$$\frac{p^{n+1}-p^{n}}{\Delta t}=-\gamma\frac{p^{n+1}}{\sum_{ijk\in V}\Delta V_{ijk}}\sum_{ijk\in V}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\Delta V_{ijk} \quad \text{(3. 136)}$$

Therefore, the pressure at the new time can be calculated as

$$p^{n+1}=p^{n}\left\{1+\frac{\gamma\Delta t}{\sum_{ijk\in V}\Delta V_{ijk}}\sum_{ijk\in V}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\Delta V_{ijk}\right\}^{-1} \quad \text{(3. 137)}$$

where, p^{n} on the right side of the equation (3. 137) is the gas pressure in the previous step, and in the case where the gas divides, it is equal to the gas pressure before division. In the case where the gas coalesces, it is set to a value obtained by weighting and averaging the pressure of the gas before coalescence by volume.



The flow velocity at the position of → is the boundary flow velocity determined by the divergence condition of the surface cell.

Figure 0-3-13 Conceptual diagram of cell division of gas region and boundary flow velocity

Chapter 4 **Contents of program**

In this chapter, contents of CADMAS-SURF/3D-MG program such as process flow and data structure are shown.

4.1. Processing flow

The processing flow of CADMAS-SURF/3D-MG is shown in

Figure 0-4-1. The processing flow of CADMAS-SURF/3D-MG consists of preprocessing for setting default values and inputting data, calculation loop for computing physical quantities at each time and outputting files, and postprocessing for closing files etc. which is roughly divided into 3 processings. The calculation loop takes the form of medium judgment iteration in order to appropriately perform file output of various physical quantities including the initial value.

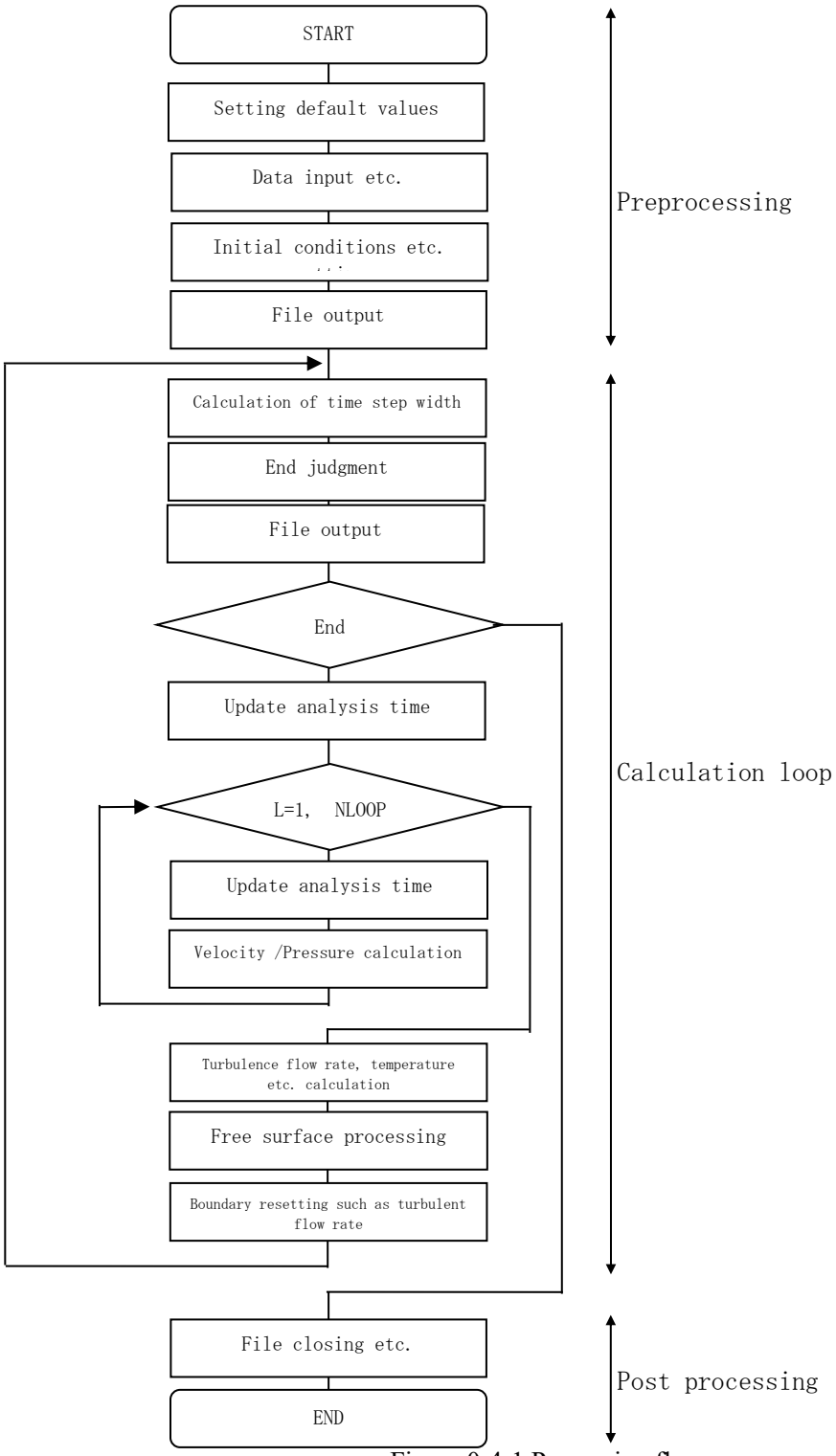


Figure 0-4-1 Processing flow

4.2. Parallelization policy and communication area

4.2.1. Parallelization policy

Considering the ease of maintenance and function expansion, it is not a good idea to have two versions of a nonparallelized program (standalone version) and a parallelized program (parallel version). Therefore, we masked the MPI which is the message exchange library, and made it possible to become a standalone version simply by invalidating (commenting out) the calling part of MPI's subroutine and include file. Of course, parallel version also operates with one processor, in that case, it made to behave like the standalone version. The image of the MPI mask is shown in Figure 0-4-2.

Since CADMAS - SURF / 3D is specialized for wave calculation, special processing is done in the vertical direction (z direction) at wave - forming boundary and radiation boundary. Therefore, the region division is performed only in the horizontal direction (x direction and y direction).

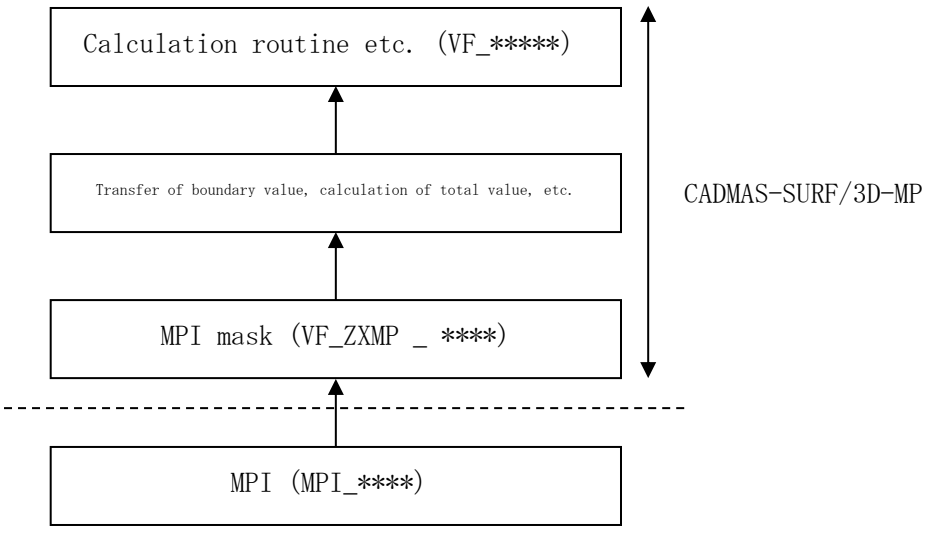


Figure 0-4-2 MPI mask image

4.2.2. Coverage area and communication area of each processor

In the region segmentation method, each processor places on the memory a region where its own calculation region and its communication region are combined. The width (layer) of the communication area is determined by the reference relation of the calculation scheme. For example, in the case of the first order upwind difference, only one layer may be used since only the relationship between the self and the adjacent cell is used. However, since the VOF method requires information next to two by the donor acceptor method, CADMAS-SURF/3D-MG decided to perform two layers of communication (see Figure 0-4-3, Figure 0-4-4).

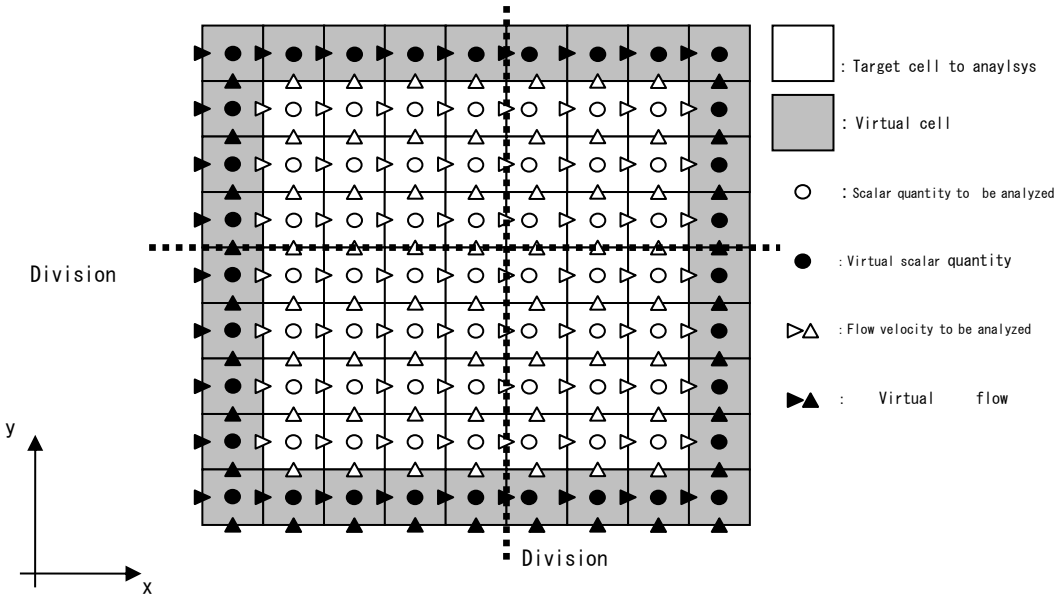


Figure 0-4-3 Definition position of whole area and variable

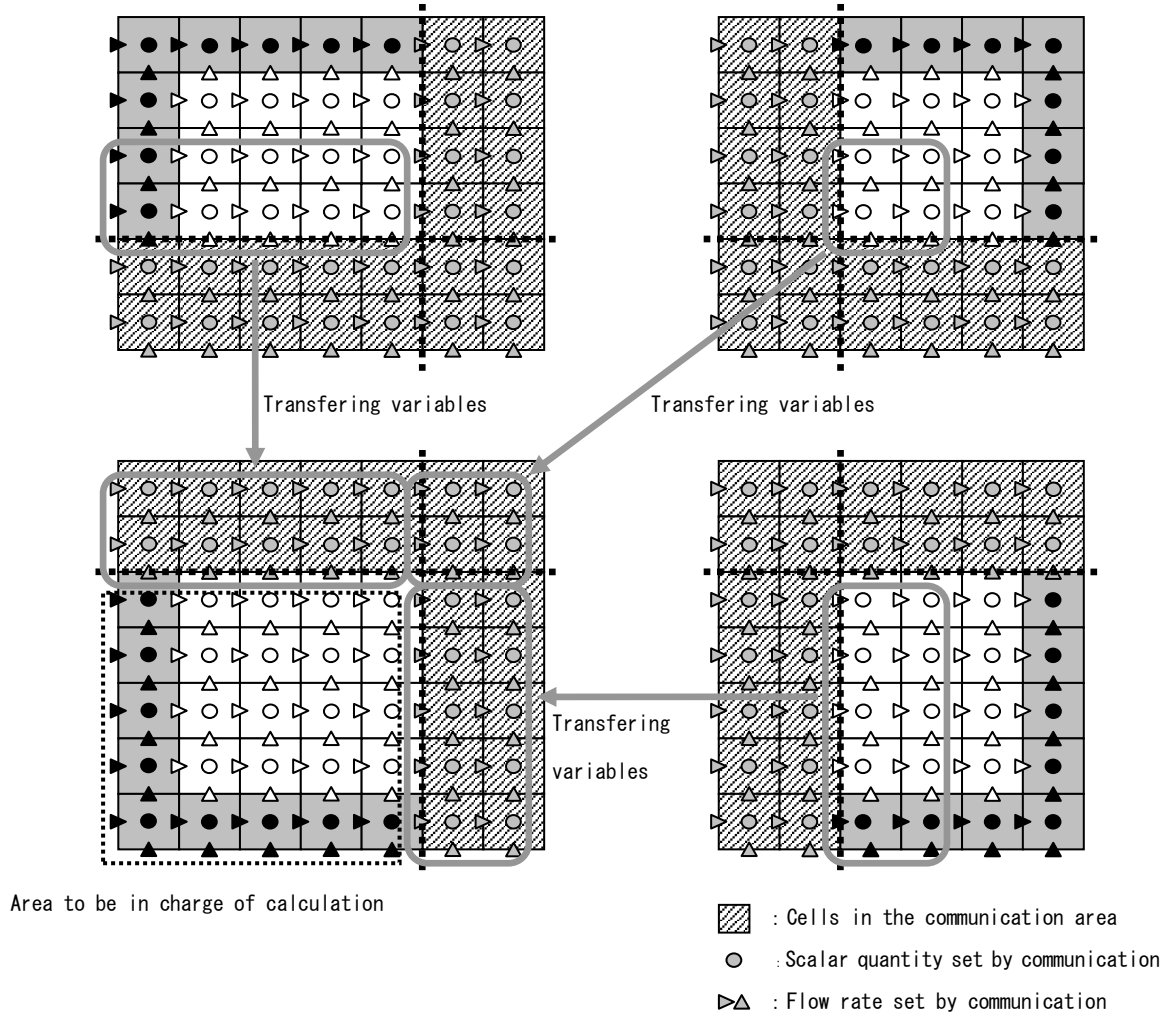


Figure 0-4-4 The assigned area and communication area after division

4.3. Coding policy

4.3.1. File classification

The classification of file names is shown in Table 0-4-1. The files that make up CADMAS-SURF/3D-MG consist of two sections, an include file and a source file, and the first 4 characters of the file name are different depending on the classification and content.

Table 0-4-1 File classification

First 4 characters	Classification	Description
VF_A	Include	Parameter statement and common sentence
vf_a	Source	Main routine and control related
vf_b	Source	Boundary condition relation
vf_c	Source	Various calculations and settings
vf_d	Source	(unused)
vf_e	Source	(unused)
vf_f	Source	Calculation of VOF function F
vf_g	Source	(unused)
vf_h	Source	(unused)
vf_i	Source	File input related
vf_j	Source	(unused)
vf_k	Source	Calculation of k - ε 2 equation model
vf_l	Source	(unused)
vf_m	Source	Solutions of simultaneous linear equations
vf_n	Source	(unused)
vf_o	Source	File output related
vf_p	Source	Transfer of boundary value for parallel, calculation of total value, etc.
vf_q	Source	(unused)
vf_r	Source	(unused)
vf_s	Source	Calculate concentration
vf_t	Source	Calculation of temperature
vf_u	Source	(unused)
vf_v	Source	Calculation of flow velocity and pressure
vf_w	Source	Calculation of wave form, wave velocity etc
vf_x	Source	(unused)
vf_y	Source	(unused)
vf_z	Source	Generic Subroutine VF_ZXMP _ **** is an MPI mask

4.3.2. Include file

The coding style of the include file is shown in Table 0-4-2. Parameter statement and common sentence are described in the include file, the type of the common variable is the global type described at the beginning of the source file (integer is 4 bytes, Real number is 8 bytes.) The outline of the include file and the contents of each variable are described as comments in the file.
Using an include file makes it possible to avoid inconsistencies in common variables, inadvertent changes in variable names, and the like.

Table 0-4-2 Include file coding style

Coding style	Description
PARAMETER (@@@@@@=@@@, @@@@@@=@@@) CD=== Overview ========== CDT VF_A0PRM. h: Files collecting PARAMETER statements C==== Contents ========== CD @@@@@@ : PRM : I*4 : @@@@@@@@@@@@@@ CD @@@@@@ : PRM : R*8 : @@@@@@@@@@@@@@	Parameter statement Overview Contents PRM：Parameter
COMMON /VF_AXXXXI/ @@@@@@, @@@@@@, & @@@@@@, @@@@@@ CD=== Overview ========== CDT VF_AXXXXI. h:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@	Common statement (Integer) Overview

CD (1)@@@@@@@@@@@@@@@@@@@@ CD (2)@@@@@@@@@@@@@@@@@@@@ C==== Contents ========== CD @@@@@@ : CNS : I*4 : @@@@@@@@@@@@@ CD @@@@@@ : TRN : I*4 : @@@@@@@@@@@@@	Contents CNS：Not changed after preprocessing TRN：Changed after preprocessing
COMMON /VF_AXXXXR/ @@@@@@, @@@@@@, & @@@@@@, @@@@@@ CD=== Overview ========== CDT VF_AXXXXR. h:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ CD (1)@@@@@@@@@@@@@@@@@@@@ CD (2)@@@@@@@@@@@@@@@@@@@@ C==== Contents ========== CD @@@@@@ : CNS : R*8 : @@@@@@@@@@@@@ CD @@@@@@ : TRN : R*8 : @@@@@@@@@@@@@	Common statement (Real) Overview Contents CNS：Not changed after preprocessing TRN：Changed after preprocessing

4.3.3. Source file coding policy

The coding style of the source file is shown in Table 0-4-3. At the beginning of the source file, a program statement or subroutine statement is described, and immediately below it, an outline of the routine is described by a comment statement.

Next, describe various declaration statements in the order of global type, global variables (capturing common variables with include statements), arguments, local variables (only when declarations are needed), especially with reference to arguments (IN).

Describe assignment (OUT) and type etc in comment text. Next, it describes in the order of executable statement, file related error processing (skipping of processing is necessary), format statement, and termination processing.

Table 0-4-3 Source file coding policy

Coding style	Description
SUBROUTINE VF_XXXXXX (@@@@@@)	Subroutine statement
CD === Overview ========== ========== CDT VF_XXXXXX: @@@@@@@@ @@@@@@@@ @ @ @ @ @ @ @ @ @ @ @ CD (1) @@@@@@@ CD (2) @@@@@@@ @	Description of Overview Various declarative statements
C ==== Declaration ========== ========== C - Global type - IMPLICIT INTEGER (I - N), DOUBLE PRECISION (A - H, O - Z)	Declaration of global type Declaring global variables
C - global variable - INCLUDE 'VF_A @@@@@.. H'	Argument description and declaration

CD - Argument -	
CD @@@@: IN: C * (*): @@@@ @ @ @ @ @ @ @ @ @ @ @	
CD @@@@: OUT: I * 4: @@@@ @ @ @ @ @ @ @ @ @ @ @	
CD @@@ @: OUT: R * 8: @@ @ @ @ @ @ @ @ @ @ @ @ @	
DIMENSION @@@@@@ @	Description and declaration of local variables (only when necessary)
CD - Local variable -	
CD @@@@: C * (*): @@ @ @ @ @ @ @ @ @ @ @ @ @	
CD @@@ @: R * 8: @@@@ @ @ @ @ @ @ @ @ @ @ @	
DIMENSION @@@@@@ @	
C ==== Execution ==========	Start of execution statement
CD - @@@@ @@ -	
CD - @@@@ @@ -	End of executable statement
C - End of executable statement -	
9000 CONTINUE	
GOTO 9999	File related error handling (only when necessary)
C ==== File related error handling ==========	
9010 CONTINUE	
CALL A2ERR ('@@@@@@', 'I / O ERROR.')	
GOTO 9999	
C ==== Format Statement ========== ======	Format statement (only when necessary)
9510 FORMAT (", '@@@@@@@@@ @@@@@@@@@@ @ @ @ @ @ @ @ @ @ @)	
C ==== Termination ==========	
9999 CONTINUE	End processing
RETURN	
END	

4.3.4. End of program execution

Stop statements for terminating program execution are described in the main routine VF_A1MAIN and VF_A2ERR which outputs an error message.Therefore, as shown in Figure 0-4-5, when the program ends normally, VF_A1MAIN Stop sentence of VF_A2ERR is executed when the stop statement of abnormal termination (such as an input error which can be detected by the program) is abnormal termination.

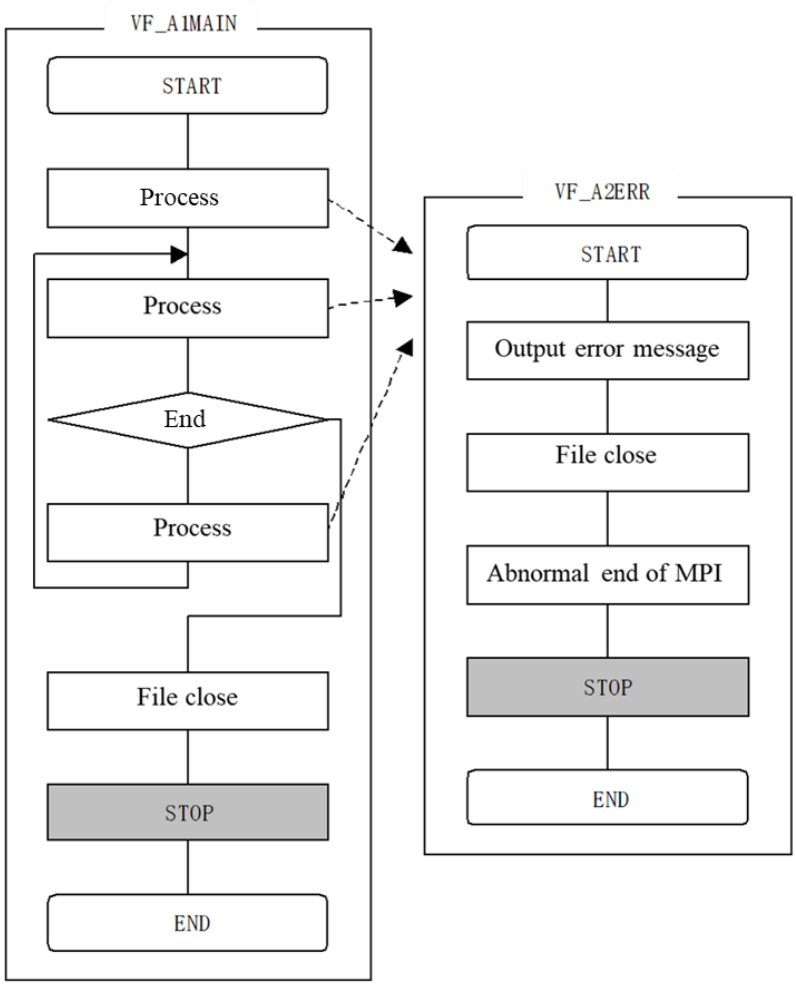


Figure 0-4-5 End of program execution

4.4. Data structure list and detailed explanation

4.4.1. Data structure list

The global data used by CADMAS-SURF/3D-MG is roughly classified as follows.

(1) Global variable with common

(2) Substantial global variable declared by the main routine

Table 0-4-4 shows a list of data structures such as common names.

Table 0-4-4 Data structure list

Name	Type	File name	Overview
—	constant	VF_A0PRM. h	Files collecting PARAMETER statements
VF_ACOMPI	common	VF_ACOMPI. h	Numerical solution related (various parameters, number of iterations, etc.): integer
VF_ACOMPR	common	VF_ACOMPR. h	Numerical solution related (various parameters and number of iterations etc.): real number
VF_ACPUTR	constant common	VF_ACPUTR. h	CPU time measurement related (CPU time etc.): Parameter and real number
VF_ADBGI	common	VF_ADBGI. h	Debug related: integer
VF_ADBGR	common	VF_ADBGR. h	Debug related: Real number
VF_AFILEI	common	VF_AFILEI. h	File related (file number, output control, etc.): integer
VF_AFILER	common	VF_AFILER. h	File relation (Output time interval etc): real number
VF_ANUMBI	common	VF_ANUMBI. h	Data number related (array size and number of grids, etc.): integer

VF_APARAI	common	VF_APARAI. h	Parallelization related: integer
VF_APARAR	common	VF_APARAR. h	Parallelization related: Real number
VF_APHYSI	common	VF_APHYSI. h	Physical event related (physical quantity and physical model etc): integer
VF_APHYSR	common	VF_APHYSR. h	Physical event related (physical quantity and physical model, etc.): real number
VF_ATIMEI	common	VF_ATIMEI. h	Time control related (analysis time and time step width etc): integer
VF_ATIMER	common	VF_ATIMER. h	Time control related (analysis time and time step width etc): real number

4.4.2. Detailed explanation of global variables by common

Table 0-4-5 shows a detailed explanation of the global variable by common. Most of the common global variable is a single variable.

Table 0-4-5 Detailed explanation of global variables by common (1/15)

File name：VF_A0PRM. h		File contains PARAMETER statement	
Variable name	Type	Variable type	Contents
IVR001	PRM	I*4	First digit of version
IVR002	PRM	I*4	The second digit of the version
MAXNPI	PRM	I*4	Maximum number of processes in x direction (for parallel use)
MAXNPJ	PRM	I*4	Maximum number of processes in y direction (for parallel use)
MAXBUF	PRM	I*4	Number of buffer data (for parallel)
MFILIN	PRM	I*4	File number of input file
MFILMT	PRM	I*4	File number of matrix data file
MFILRE	PRM	I*4	File number of the restart file
MFILPR	PRM	I*4	File number of time dependent porosity file
MFILLP	PRM	I*4	File number of list file
MFILGR	PRM	I*4	File number of the chart file
MFILRS	PRM	I*4	File number of detail file
MFILTR	PRM	I*4	File number of time series file
MAXCHR	PRM	I*4	Maximum number of characters per line (input file)
MAXWDS	PRM	I*4	Maximum number of words per line (input file)
MAXG1	PRM	I*4	First array size of XX, YY and ZZ
MAXB1	PRM	I*4	First array size of INDB
MAXNC	PRM	I*4	Maximum number of components of concentration
MAXTR	PRM	I*4	Maximum number of output target data to time series file
MAXTR1	PRM	I*4	First array size of ITRPRM
MAXPRB	PRM	I*4	Maximum number of spatial blocks of time dependent porosity
MAXPVC	PRM	I*4	Maximum number of bubbles (for calculating air pressure)
MAXDRG	PRM	I*4	Maximum number of coefficients of Dupuit-Forheimer expression
ZERO	PRM	R*8	Zero judgment value
ZEROG	PRM	R*8	Zero judgment value (for judgment of grid spacing)
PI	PRM	R*8	Pi

Table 0-4-6 Detailed explanation of global variables by common (2/15)

File name：VF_ACOMPI.h		Numerical solution related to parameters, number of iterations, etc, (Integer)	
Variable name	Type	Variable type	Contents
ICGTYP	CNS	I*4	Types of preprocessing for solving simultaneous linear equations = 0: Incomplete LU factorization (ILU) ! = 0: Corrected incomplete LU factorization (MILU)
ICGMAX	CNS	I*4	Maximum number of repetitions of solving simultaneous linear equations
ICGITR	TRN	I*4	Number of repetitions of solving simultaneous linear equations
ISCMVP	CNS	I*4	Difference scheme of convection term of flow velocity = 0: DONOR ! = 0: Higher order scheme (undeveloped)
ISCMFF	CNS	I*4	Difference scheme of advection term of VOF function F = 0: donor-acceptor method ! = 0: Method considering the inclination of the interface
ISCMK	CNS	I*4	Difference scheme of advection term of k - ε: same ISCMVP
ISCMT	CNS	I*4	Difference scheme of advection term of temperature: same ISCMVP
ISCMC(MAXNC)	CNS	I*4	Difference scheme of advection term of concentration: same ISCMVP
IBSUW0	CNS	I*4	Flow velocity calculation method of surface cell = 0: linear extrapolation if extrapolation from the fluid side is possible ! = 0: Gradient zero

Table 0-4-7 Detailed explanation of global variables by common(3/15)

File name：VF_ACOMPR.h		Numerical solution related to parameters, number of iterations, etc, (Real)	
Variable name	Type	Variable type	Contents
CGPARA	CNS	R*8	Parameters for MILU of solving simultaneous linear equations

CGEPSA	CNS	R*8	Convergence determination value (absolute error) of solving method of simultaneous linear equation
CGEPSR	CNS	R*8	Convergence judgment value (relative error) of solving method of simultaneous linear equation
CGDIV	CNS	R*8	Divergence judgment value of solving method of simultaneous linear equation
CGBNRM	TRN	R*8	Norm on the right side of simultaneous linear equations solution
CGXNRM	TRN	R*8	Norm of the residuals of solving simultaneous linear equations
FEPS	CNS	R*8	Zero judgment value of VOF function F
FLOWER	CNS	R*8	Lower limit of VOF function F
FUPPER	CNS	R*8	Upper limit of VOF function F
FSUM	TRN	R*8	Spatial integral value of VOF function F
FCUT	TRN	R*8	Cutoff value of VOF function F
PLOWER	CNS	R*8	The lower limit value of the porous value
SCMVP	CNS	R*8	Parameters of the DONOR scheme (0 0 - 1 0: for flow rate) = 0. 0: Secondary center = 1. 0: 1st order windward
SCMK	CNS	R*8	Parameters of DONOR scheme (for k - ε): same as SCMVP
SCMT	CNS	R*8	DONOR scheme parameters (for temperature): same as SCMVP
SCMC(MAXNC)	CNS	R*8	DONOR scheme parameters (for concentration): same as SCMVP

Table 0-4-8 Detailed explanation of global variables by common(4/15)

File name：VF_ACPUTR. h		CPU time measurement related to CPU time etc. (Parameter and real number)	
Variable name	Type	Variable type	Contents
ICPUIN	PRM	I*4	Timer processing flag (initialization)
ICPUST	PRM	I*4	Timer processing flag (start)
ICPUEN	PRM	I*4	Timer processing flag (stop, take a total)
ICPUOU	PRM	I*4	Timer processing flag (CPU time output)
KCP0AL	PRM	I*4	Timer type flag (doubles as the total number flag)
KCP***	PRM	I*4	Timer type flag (various processing)
CPUS(KCP0AL)	TRN	R*8	Total CPU time
CPUW(KCP0AL)	TRN	R*8	Time when the timer was started

Table 0-4-9 Detailed explanation of global variables by common(5/15)

File name：VF_ADBGI. h		Debug (Integer)	
Variable name	Type	Variable type	Contents
IDBGF(6)	CNS	I*4	F value specification data to rectangular box (1): x direction cell number of the starting point (2): cell number in the y direction of the starting point (3): z-direction cell number of starting point (4): x direction cell number of the end point (5): Y direction cell number of the end point (6): Z direction cell number of the end point
IDBGTD(6)	CNS	I*4	Speed specification data for TD to rectangular box (1): x direction cell number of the starting point (2): cell number in the y direction of the starting point (3): z-direction cell number of starting point (4): x direction cell number of the end point (5): Y direction cell number of the end point (6): Z direction cell number of the end point

Table 0-4-10　Detailed explanation of global variables by common(6/15)

File name：VF_ADBGR. h		Debug (Real)	
Variable name	Type	Variable type	Contents
RDBGF	CNS	I*4	F value specification data (F value) to rectangular box
RDBGTD(3)	CNS	I*4	Speed specification data for TD to V rectangular box (V value) (1): x direction velocity (2): y direction velocity (3): z direction velocity

Table 0-4-11　Detailed explanation of global variables by common(7/15)

File name：VF_AFILEI. h(1/3)		File related to file number and output control(Integer)	
Variable name	Type	Variable type	Contents
IINFIL	CNS	I*4	File number of input file (0 = not open)
IMTFIL	CNS	I*4	File number of matrix data file
IREFIL	CNS	I*4	File number of the restart file
IPRFIL	CNS	I*4	File number of time dependent porosity file
ILPFIL	CNS	I*4	File number of list file
IGRFIL	CNS	I*4	File number of the chart file
IRSFIL	CNS	I*4	File number of detail file
ITRFIL	CNS	I*4	File number of time series file
ILPTYP	CNS	I*4	How to output the list file = 0: Do not output = 1: step interval = 2: analysis time interval
ILPTRN(3)	CNS	I*4	List file output step information (1): output start step (2): output end step (3): output step interval
ILPARA(3)	CNS	I*4	Output section information of the list file (1): Output section = 1: xy section, = 2: xz section, = 3: yz section = 4: yx section, = 5: zx section, = 6: zy section

			(2): cell number of output section (3): grid number of output section
ILPON(21)	CNS	I*4	Output / non-output to list file (1): Obstacle = 0: No output,! = 0: Output (2): Index on boundary condition = 0: No output,! = 0: Output (3): Inertial force coefficient = 0: No output,! = 0: Output (4): resistance coefficient = 0: No output,! = 0: Output (5): Porous value = 0: No output,! = 0: Output (6): GLV, GLX, GLZ = 0: No output,! = 0: Output (7): NF = 0: No output,! = 0: Output (8): VOF function F = 0: No output,! = 0: Output (9): flow velocity = 0: No output,! = 0: Output (10): Pressure = 0: No output,! = 0: Output (11): sum of molecular kinematic viscosity coefficient and vortical viscosity coefficient = 0: No output,! = 0: Output (12): boundary value = 0: No output,! = 0: Output (13): temperature = 0: No output,! = 0: Output (14): Thermal conductivity = 0: No output,! = 0: Output (15): Concentration = 0: No output,! = 0: Output (16): diffusion coefficient

			= 0: No output,! = 0: Output (17): Turbulent energy = 0: No output,! = 0: Output (18): Turbulent energy dissipation = 0: No output,! = 0: Output (19): Vortical viscosity coefficient = 0: No output,! = 0: Output (20): index for calculation of air pressure = 0: No output,! = 0: Output (21): Pressure for calculation of air pressure = 0: No output,! = 0: Output
IGRTYP	CNS	I*4	How to output the chart file = 0: Do not output = 1: step interval = 2: analysis time interval
IGRTRN(3)	CNS	I*4	Output step of graphic file Step information (1): output start step (2): output end step (3): output step interval
IGRARA(6)	CNS	I*4	Output section information of the list file (1): x direction cell number of the starting point (2): cell number in the y direction of the starting point (3): z-direction cell number of starting point (4): x direction cell number of the end point (5): Y direction cell number of the end point (6): Z direction cell number of the end point
IGRVOR	CNS	I*4	Whether to output vorticity to a chart file = 0: Do not output ! = 0: Output
IRSTYP	CNS	I*4	How to output detailed file = 0: Do not output = 1: step interval = 2: analysis time interval
IRSTRN(3)	CNS	I*4	Detailed file output step information

			(1): output start step (2): output end step (3): output step interval
IRETYP	CNS	I*4	How to restart <0: Do not restart > = 0: Restart step
ITRTYP	CNS	I*4	Output method of time series file = 0: Do not output = 1: step interval = 2: analysis time interval
ITRTRN(3)	CNS	I*4	Time series file output step information (1): output start step (2): output end step (3): output step interval
ITRNUM	CNS	I*4	Number of data to be output to time series file
ITRPRM(MAXTR1, MAXTR)	CNS	I*4	Parameters of time series file
			(1, *): How to calculate the value = 0: Water level fluctuation from initial water level = 1: value of specified point = 2: Minimum value of specified area = 3: Maximum value of specified area = 4: Volume average value of specified area = 5: Volume integral value of specified area = 11: Wave power to obstacles (2, *): Type of physical quantity <0: Concentration (component number × (-1)) = 0: Water level fluctuation from initial water level = 1: x direction flow velocity = 2: y direction flow velocity = 3: z direction flow velocity = 4: Pressure = 5: F value = 6: Turbulent energy

			= 7: Turbulent energy dissipation = 8: temperature = 11: Wave force in the negative direction of x to an obstacle = 12: Wave force in the positive direction of x to the obstacle = 13: Wave force in the negative direction of y to the obstacle = 14: positive wave force of y to the obstacle = 15: negative wave force of z to obstacles = 16: positive wave force of z to the obstacle = 17: vorticity (x direction component) = 18: vorticity (y direction component) = 19: vorticity (z direction component) (3, *): x direction grid or cell number (I 1) = -1: Solve the wave boundary at the x coordinate minimal position = -2: Solve the wave boundary at the x-coordinate maximum position = -3: Solve the wave boundary at y coordinate minimal position = -4: Solve the wave boundary at y coordinate maximum position (4, *): y direction grid or cell number (J 1) (5, *): z direction grid or cell number (K 1) (6, *): x direction grid or cell number (I 2) (7, *): y direction grid or cell number (J 2) (8, *): z direction grid or cell number (K 2)
IPRNT	CNS	I*4	Number of data of the porosity in the time direction = 0: Do not read the porosity file = 1: It is not time dependent data > 1: Time dependent data
IPRNB	CNS	I*4	Maximum number of spatial blocks of time dependent porosity

IPRNP	CNS	I*4	Number of cells with time dependent porosity setting
IPRARA(6, MAXPRB)	CNS	I*4	Time-dependent porosity spatial block information (1, *): x direction cell number of start point (2, *): Y direction cell number of start point (3, *): Z direction cell number of start point (4, *): x direction cell number of the end point (5, *): Y direction cell number of end point (6, *): Z direction cell number of the end point
IPRIT	TRN	I*4	Current time block being read
MTBTYP	CNS	I*4	Matrix data type = 0: Not used = 1: Water level and flow rate specification = 2: flow velocity specification = 3: Water level specification
MTBTT	CNS	I*4	Number of data in the phase direction of matrix data
MTBZZ	CNS	I*4	Number of data in the water depth direction of the matrix data
MTBNOW	TRN	I*4	Current position in the phase direction of matrix data

Table 0-4-12 Detailed explanation of global variables by common(8/15)

File name：VF_AFILER. h		File related to file number and output control(Real)	
Variable name	Type	Variable type	Contents
RLPTRN(4)	TRN	R*8	Output time information of the list file (1): output start time (2): output end time (3): Output time interval (4): Next output time
RGRTRN(4)	TRN	R*8	Output time information of the chart file

			(1): output start time (2): output end time (3): Output time interval (4): Next output time
RRSTRN(4)	TRN	R*8	Detailed file output time information (1): output start time (2): output end time (3): Output time interval (4): Next output time
RTRTRN(4)	TRN	R*8	Output time information of time series file (1): output start time (2): output end time (3): Output time interval (4): Next output time
RTRVAL(MAXTR)	TRN	R*8	Calculated value output to time series file
PRTOLD	TRN	R*8	Time of the previous porosity block
PRTNOW	TRN	R*8	Time of the current porosity block
DMTBT0	CNS	R*8	Initial dimensionless phase of matrix data

Table 0-4-13 Detailed explanation of global variables by common(9/15)

File name：VF_ANUMBI. h		Data number related t Array size and grid number (Integer)	
Variable name	Type	Variable type	Contents
NUMI	CNS	I*4	Number of x direction grids + 1 (number of x direction cells + 2)
NUMJ	CNS	I*4	Number of grids in y direction + 1 (number of cells in y direction + 2)
NUMK	CNS	I*4	Number of z direction grids + 1 (number of z direction cells + 2)
NUMB	CNS	I*4	Number of boundary surfaces
NUMS	TRN	I*4	Number of surface cells
LEQK	CNS	I*4	Whether k - ε is calculated or not = 0: Do not calculate ! = 0: Calculate
LEQT	CNS	I*4	Whether or not to calculate the temperature = 0: Do not calculate

			! = 0: Calculate
LEQC	CNS	I*4	Whether to calculate the concentration = 0: Do not calculate > = 1: number of components
NUMIJ	CNS	I*4	Horizontal grid number for wave generation source + 1

Table 0-4-14 Detailed explanation of global variables by common(10/15)

File name：VF_APARAI. h		Parallelization related(Integer)	
Variable name	Type	Variable type	Contents
NPROCS	CNS	I*4	Number of processes
NUMNPI	CNS	I*4	x direction process number
NUMNPJ	CNS	I*4	Number of y direction process
MYRANK	CNS	I*4	My rank
MYRI	CNS	I*4	Your x direction rank
MYRJ	CNS	I*4	Your y direction rank
NUMI0	CNS	I*4	The total number of grids in x direction + 1
NUMJ0	CNS	I*4	The total number of grids in the y direction + 1
MYIS	CNS	I*4	x direction cell number (start, virtual not included, local)
MYIE	CNS	I*4	x direction cell number (end, virtual not included, local)
MYJS	CNS	I*4	y direction cell number (start, virtual not included, local)
MYJE	CNS	I*4	Y direction cell number (end, virtual not included, local)
MYMIS	CNS	I*4	x direction cell number (start, virtual thickness)
MYMIE	CNS	I*4	x direction cell number (end, virtual thickness)
MYMJS	CNS	I*4	Y direction cell number (start, virtual thickness)
MYMJE	CNS	I*4	Y direction cell number (end, virtual thickness)
MYGIS	CNS	I*4	x direction cell number (start, virtual inclusion, global)
MYGIE	CNS	I*4	x direction cell number (end, virtual inclusive, global)
MYGJS	CNS	I*4	Y direction cell number (start, virtual inclusion, global)
MYGJE	CNS	I*4	Y direction cell number (end, virtual inclusive, global)
NUMBUF	CNS	I*4	Number of buffer data (1 piece)
IPROCS(0:MAXNPI)	CNS	I*4	End cell number in x direction
JPROCS(0:MAXNPJ)	CNS	I*4	End cell number in y direction

Table 0-4-15 Detailed explanation of global variables by common(11/15)

File name：VF_APARAR. h		Parallelization related(Real)	
Variable name	Type	Variable type	Contents
GLXMIN	CNS	R*8	The minimum value of the overall x direction grid coordinate
GLXMAX	CNS	R*8	The maximum value of the overall x direction grid coordinate
GLYMIN	CNS	R*8	The minimum value of the overall y direction grid coordinate
GLYMAX	CNS	R*8	The maximum value of the overall y direction grid coordinate

Table 0-4-16 Detailed explanation of global variables by common(12/15)

File name：VF_APHYSI. h		Physical event related to physical quantity, physical model, etc(Integer)	
Variable name	Type	Variable type	Contents
IBCTYP(2, 4)	CNS	I*4	Information on special boundaries (*, 1): boundary of x coordinate minimum position (*, 2): boundary of x coordinate maximum position (*, 3): boundary of y coordinate minimum position (*, 4): boundary of y coordinate maximum position (1, *): Type of special boundary = 0: None = 1: Wave boundary in the normal direction = 2: Open boundary in the normal direction (2, *): Details of type of special boundary In case of wave boundary = -3: Matrix data = -2: Stokes wave (fifth order approximation solution) = -1: Cnoidal wave (third order approximation solution) = 0: Stokes wave or Cnoidal wave > 0: Flow function method B and its order In case of open boundary

			= 0: Radiation boundary (wave velocity of small amplitude wave) (3, *): Starting cell number of special boundary (4, *): End point cell number of special boundary
IDAMP(4)	CNS	I*4	Setting flag of attenuation region (1): near x-coordinate minimum position (2): near x coordinate maximum position (3): near the y coordinate minimum position (4): near y coordinate maximum position = -1: Not used > = 0: Used and the order of the decay function
IDROP	CNS	I*4	Free drop treatment of water drops (Timer Do method) = 0: Do not process = 1: Perform processing
ISCTYP(2)	CNS	I*4	Information on wave source (1): Type of wave production source = 0: None > 0: Wave source in the x direction (ABS = I) <0: Wave source in the y direction (ABS = J) (2): wave function = -3: Matrix data = -2: Stokes wave (fifth order approximation solution) = -1: Cnoidal wave (third order approximation solution) = 0: Stokes wave or Cnoidal wave > 0: Flow function method B and its order
NPVCB	TRN	I*4	Number of bubbles (for calculating air pressure)
IPVCBC(MAXPVC)	TRN	I*4	Work for calculating air pressure
IDRGN	CNS	I*4	Resistance by Dupuit-Forheimer equation = 0: Resistance is an existing process > 0: Dupuit-Forheimer equation (number of inputs of grain size)

Table 0-4-17 Detailed explanation of global variables by common(13/15)

File name：VF_APHYSR. h	Physical event related to physical quantity, physical model, etc(Real)

Variable name	Type	Variable type	Contents
UINI	CNS	R*8	Initial value of x direction flow velocity
VINI	CNS	R*8	Initial value of y direction flow velocity
WINI	CNS	R*8	Initial value of flow velocity in z direction
RHO0	CNS	R*8	density
ANU0	CNS	R*8	Molecular kinematic viscosity coefficient
GRZ0	CNS	R*8	Z component of gravitational acceleration (negative direction)
WVLVL	CNS	R*8	Initial water surface height (z coordinate)
AKMINK	CNS	R*8	Minimum value of turbulent energy
AKMINE	CNS	R*8	Minimum value of turbulent energy dissipation
AKINIK	CNS	R*8	Initial value of turbulent energy
AKINIE	CNS	R*8	Initial value of turbulent energy dissipation
AKCMU	CNS	R*8	Cμ
AKSGK	CNS	R*8	σk
AKSGE	CNS	R*8	σ e
AKC1	CNS	R*8	C1
AKC2	CNS	R*8	C2
AKC3	CNS	R*8	C3
AKK0	CNS	R*8	Logarithmic law κ
AKA0	CNS	R*8	Logarithmic rule A
AKPR	CNS	R*8	Turbulent Prandtl number
AKSM(MAXNC)	CNS	R*8	Turbulent Schmidt number
TINI	CNS	R*8	Initial value of temperature
TCP0	CNS	R*8	Constant pressure specific heat
TCN0	CNS	R*8	Thermal conductivity
TDT0	CNS	R*8	Reference temperature for buoyancy calculation
TDR0	CNS	R*8	Temperature differential of density
CINI(MAXNC)	CNS	R*8	Initial value of concentration
CDF0(MAXNC)	CNS	R*8	Diffusion coefficient
CDC0(MAXNC)	CNS	R*8	Reference concentration for buoyancy calculation
CDR0(MAXNC)	CNS	R*8	Concentration derivative of density
BCTYP(11, 4)	TRN	R*8	Information on special boundaries (*, 1): boundary of x coordinate minimum position

			(*, 2): boundary of x coordinate maximum position (*, 3): boundary of y coordinate minimum position (*, 4): boundary of y coordinate maximum position In case of wave boundary (1, *): depth of water (2, *): wave height (3, *): cycle (4, *): wavelength (5, *): Ursell number (6, *): Nondimensional phase at zero water level fluctuation (7, *): Current wave height to be produced (8, *): how many cycles to amplify (9, *): Incident angle (zero normal direction) (10, *): x coordinate value of reference point (11, *): y coordinate value of reference point
			In case of open boundary (1, *): depth of water (2, *): Dummy (3, *): cycle (4, *): wavelength (5, *): Dummy (6, *): wave velocity (7, *): Dummy (8, *): Dummy (9, *): Dummy (10, *): Dummy (11, *): Dummy
DAMP(4, 4)	CNS	R*8	Information on the attenuation region (*, 1): Near the minimum x-coordinate position (*, 2): near x coordinate maximum position (*, 3): near the y coordinate minimal position (*, 4): near y coordinate maximum position (1, *): horizontal parameter of attenuation function (2, *): Vertical parameter of attenuation function (3, *): width of attenuation region

			(4, *): Water depth of attenuation region
WBUB	CNS	R*8	Rising speed of bubble (Timer Do method) <ZERO: Do not process > = ZERO: Perform processing
SCTYP(8)	TRN	R*8	Information on wave source (1): depth of water (2): wave height (3): cycle (4): wavelength (5): Ursell number (6): Nondimensional phase at zero water level fluctuation (7): the height of the current wave to be produced (8): how many cycles to amplify
PVCP0	CNS	R*8	Atmospheric pressure (= 0.0: Do not calculate air pressure)
PVCGM	CNS	R*8	Specific heat ratio
PVCDIV(MAXPVC)	TRN	R*8	Work for calculating air pressure
PVCPES(MAXPVC)	TRN	R*8	Work for calculating air pressure
PVCPFS(MAXPVC)	TRN	R*8	Work for calculating air pressure
PVCVES(MAXPVC)	TRN	R*8	Work for calculating air pressure
PVCVFS(MAXPVC)	TRN	R*8	Work for calculating air pressure
DRGYU	CNS	R*8	Dynamic viscosity coefficient of Dupuit-Forheimer equation
DRGDR(MAXDRG)	CNS	R*8	Dupuit-Forheimer type riprapy grain size
DRGAP(MAXDRG)	CNS	R*8	The coefficient α of the Dupuit-Forheimer equation
DRGBT(MAXDRG)	CNS	R*8	The coefficient β of the Dupuit-Forheimer equation

Table 0-4-18 Detailed explanation of global variables by common(14/15)

File name：VF_ATIMEI. h		Time control related to analysis time and time step width etc(integer)	
Variable name	Type	Variable type	Contents
NEND	CNS	I*4	Analysis end step
NNOW	TRN	I*4	Analysis step
IDTTYP	CNS	I*4	Calculation method of time step width = 0: Constant ! = 0: Automatic
LOOPS	CNS	I*4	Number of sub-loops for flow velocity / pressure calculation = 1: normal calculation > 1: With sub-loop

Table 0-4-19 Detailed explanation of global variables by common(15/15)

File name：VF_ATIMER. h		Time control related to analysis time and time step width etc(real)	
Variable name	Type	Variable type	Contents
TEND	CNS	R*8	Analysis end time
TNOW	TRN	R*8	Analysis time
DTNOW	TRN	R*8	Time step width
DTCNST	CNS	R*8	Constant value of time step width
DTINIT	CNS	R*8	Initial value of time step width
DTMIN	CNS	R*8	Minimum time step width
DTMAX	CNS	R*8	Maximum value of time step width
DTSAFE	CNS	R*8	Safety rate of time step width

4.4.3. Detailed explanation of substantial global variables declared by the main routine

Table 0-4-20 shows the detailed explanation of substantial global variables declared by the main routine. The global variables declared by the main routine are all array variables (flow velocity, pressure, etc.). It is determined dynamically.

Table 0-4-20 Detailed explanation of substantial global variables declared by the main routine (1/2)

Variable name	Type	Contents	Reference
XX(MAXG1, NUMI)	R*8	x direction grid coordinate etc. (1, I): grid coordinate x (I) (2, I): dx = x (I + 1) - x (I) (3, I): cx = (dx (I) + dx (I - 1)) / 2. 0 (4, I): 1.0 / dx (5, I): 1.0 / cx (6, I): 1.0 / (dx (I) + dx (I - 1))	—
YY(MAXG1, NUMJ)	R*8	Y direction grid coordinate etc. (1, J): grid coordinate y (J) (2, J): dy = y (J + 1) - y (J) (3, J): cy = (dy (J) + dy (J-1)) / 2. 0 84, J): 1.0 / dy (5, J): 1.0 / cy (6, J): 1.0 / (dy (J) + dy (J-1))	—
ZZ(MAXG1, NUMK)	R*8	Z direction grid coordinate etc. (1, K): grid coordinate z (K) (2, K): dz = z (K + 1) - z (K) (3, K): cz = (dz (K) + dz (K - 1)) / 2. 0 (4, K): 1.0 / dz (5, K): 1.0 / cz (6, K): 1.0 / (dz (K) + dz (K - 1))	—
UU(NUMI, NUMJ, NUMK)	R*8	x direction flow velocity	Figure 0-4-6(1)
VV(NUMI, NUMJ, NUMK)	R*8	Y direction flow velocity	Figure 0-4-6(2)

WW(NUMI, NUMJ, NUMK)	R*8	z direction flow velocity	Figure 0-4-7(1)
PP(NUMI, NUMJ, NUMK)	R*8	pressure	Figure 0-4-7(2)
FF(NUMI, NUMJ, NUMK)	R*8	VOF function F	Figure 0-4-7(2)
ANU(NUMI, NUMJ, NUMK)	R*8	Sum of molecular dynamics viscosity coefficient and vortical viscosity coefficient	Figure 0-4-7(2)
CM0(NUMI, NUMJ, NUMK)	R*8	Inertial force coefficient	Figure 0-4-7(2)
CD0(NUMI, NUMJ, NUMK)	R*8	Resistance coefficient	Figure 0-4-7(2)
GGV(NUMI, NUMJ, NUMK)	R*8	Porosity	Figure 0-4-9(2)
GGX(NUMI, NUMJ, NUMK)	R*8	X-direction area transmittance	Figure 0-4-6(1)
GGY(NUMI, NUMJ, NUMK)	R*8	Y-direction area transmittance	Figure 0-4-6(2)
GGZ(NUMI, NUMJ, NUMK)	R*8	Z-direction area transmittance	Figure 0-4-7(1)
GLV(NUMI, NUMJ, NUMK)	R*8	= GGV + (1 - GGV) * CM	Figure 0-4-7(2)
GLX(NUMI, NUMJ, NUMK)	R*8	= GGX + (1 - GGX) * CM	Figure 0-4-6(1)
GLY(NUMI, NUMJ, NUMK)	R*8	= GGY + (1 - GGY) * CM	Figure 0-4-6(2)
GLZ(NUMI, NUMJ, NUMK)	R*8	= GGZ + (1 - GGZ) * CM	Figure 0-4-7(1)
BCU(NUMB)	R*8	boundary value of x direction flow velocity	Figure 0-4-9(2)
BCV(NUMB)	R*8	Boundary value of flow velocity in y direction	Figure 0-4-9(2)
BCW(NUMB)	R*8	Boundary value of flow velocity in z direction	Figure 0-4-9(2)

BCP(NUMB)	R*8	Boundary value of pressure	Figure 0-4-9(2)
BCF(NUMB)	R*8	Boundary value of VOF function F	Figure 0-4-9(2)
BCVI(NUMB)	R*8	Flow velocity boundary condition (wall surface roughness)	Figure 0-4-9(2)
TBUB(NUMK)	R*8	Time when the bubble rising process was finally performed	—
DROPTX(NUMI, NUMJ, NUMK)	R*8	Time at which free-fall processing was last performed (x)	Figure 0-4-7(2)
DROPTY(NUMI, NUMJ, NUMK)	R*8	Time at which free-fall processing was last performed (y)	Figure 0-4-7(2)
DROPTZ(NUMI, NUMJ, NUMK)	R*8	Time at which free-fall processing was last performed (z)	Figure 0-4-7(2)
DROPUU(NUMI, NUMJ, NUMK)	R*8	Fall falling x direction velocity	Figure 0-4-7(2)
DROPVV(NUMI, NUMJ, NUMK)	R*8	Y direction velocity of free fall	Figure 0-4-7(2)
DROPWW(NUMI, NUMJ, NUMK)	R*8	Z direction velocity of free fall	Figure 0-4-7(2)
GGVOLD(IPRNP)	R*8	Porosity of previous time block	—
GGVNOW(IPRNP)	R*8	Porosity of current time block	—
GGV0(NUMI, NUMJ, NUMK)	R*8	Porosity (for time dependence)	Figure 0-4-7(2)
GLV0(NUMI, NUMJ, NUMK)	R*8	= GGV + (1 - GGV) * CM (for time dependence)	Figure 0-4-7(2)
ANUT(NUMI, NUMJ, NUMK)	R*8	Vortical viscosity coefficient ν t	Figure 0-4-7(2)
AK(NUMI, NUMJ, NUMK)	R*8	Turbulent energy	Figure 0-4-7(2)
AE(NUMI, NUMJ, NUMK)	R*8	Turbulent energy dissipation	Figure 0-4-7(2)
BCK(NUMB)	R*8	Boundary value of turbulent energy	Figure 0-4-9(2)
BCE(NUMB)	R*8	Boundary value of turbulent energy dissipation	Figure 0-4-9(2)
TT(NUMI, NUMJ, NUMK)	R*8	temperature	Figure 0-4-7(2)

ALM(NUMI, NUMJ, NUMK)	R*8	Sum of thermal conductivity and turbulent thermal conductivity	Figure 0-4-7(2)
BCT(NUMB)	R*8	Temperature boundary value	Figure 0-4-9(2)
BCTI(2, NUMB)	R*8 R*8 R*8	Temperature boundary condition (1, L): heat transfer coefficient or heat flux (2, L): external temperature	Figure 0-4-9(2)
CC(NUMI, NUMJ, NUMK, LEQC)	R*8	concentration	Figure 0-4-7(2)
DD(NUMI, NUMJ, NUMK, LEQC)	R*8	Sum of diffusion coefficient and turbulent diffusion coefficient	Figure 0-4-7(2)
BCC(NUMB, LEQC)	R*8	Boundary value of concentration	Figure 0-4-9(2)
BCCI(2, NUMB, LEQC)	R*8 R*8 R*8	Concentration boundary condition (1, L): mass transfer coefficient or diffusion flux (2, L): external concentration	Figure 0-4-9(2)
DMTBTT(MTBTT)	R*8	Nondimensional phase of matrix data	—
DMTBZZ(MTBZZ)	R*8	The z coordinate of the matrix data (average water level is zero)	—
DMTBHH(MTBTT)	R*8	Water level of matrix data	—
DMTBUN(MTBZZ, MTBTT)	R*8	Horizontal flow velocity of matrix data	—
DMTBUT(MTBZZ, MTBTT)	R*8	Vertical flow velocity of matrix data	—
DBUF(NUMBUF*MAXBUF)	R*8	Buffer for parallel	—
SRCUV(NUMIJ, NUMK)	R*8	Horizontal flow velocity for wave source	—
PPPVC(NUMI, NUMJ, NUMK)	R*8	Pressure for calculating air pressure (pressure in E, S cell)	—
WK01-17(NUMI, NUMJ, NUMK)	R*8	Work array	—
WKBC(NUMB)	R*8	Work array	—

Table 0-4-21 Detailed explanation of substantial global variables declared by the main routine (2/2)

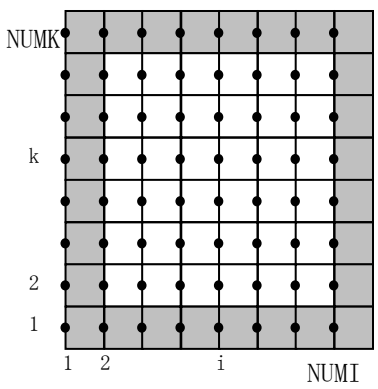
Variable name	Type	Contents	Reference
NF(NUMI, NUMJ, NUMK)	I*4	Index indicating the state of the cell = -1: Obstacle cell = 0: fluid cell = 1: Surface cell: x Fluid in negative direction = 2: Surface cell: x Fluid in positive direction = 3: Surface cell: y Fluid in negative direction = 4: Surface cell: y Fluid in positive direction = 5: Surface cell: z Fluid in negative direction = 6: Surface cell: z Fluid in positive direction = 8: Gas cell	Figure 0-4-7(2)
INDX(NUMI, NUMJ, NUMK)	I*4	Index indicating the state of the x plane = -1: Obstacle surface = 0: normal plane > = 1: Boundary (pointer to INDB)	Figure 0-4-8(1)
INDY(NUMI, NUMJ, NUMK)	I*4	Index showing the state of y plane = -1: Obstacle surface = 0: normal plane > = 1: Boundary (pointer to INDB)	Figure 0-4-8(2)
INDZ(NUMI, NUMJ, NUMK)	I*4	Index indicating the state of z plane = -1: Obstacle surface	Figure 0-4-9(1)

		= 0: normal plane > = 1: Boundary (pointer to INDB)	
INDC(NUMI, NUMJ, NUMK)	I*4	Index indicating the calculation state of the cell = -1: Non-calculated cell (obstacle or gas) = 0: calculation cell (fluid or surface)	Figure 0-4-10
INDB(MAXB1, NUMB)	I*4	Boundary index (1, L): One-dimensional representation of the I, J, K coordinates of the boundary surface (I + NUMI * (J - 1) + NUMI * NUMJ * (K - 1)) (2, L): Orientation direction = 1: Structure on negative side of x direction = 2: Structure on the x-direction positive side = 3: Structure on negative side in y direction = 4: Structure on the positive side in y direction = 5: structure on the negative side in the z direction = 6: Structure on the positive side in z direction (3, L): Boundary conditions of flow velocity / pressure = 0: undefined = 1: Slip = 2: Non slip = 3: Fixed flow rate = 4: Free = 5: wave boundary = 6: logarithmic law	Figure 0-4-9(2)

		= 7: Radiation boundary = 8: Perfect rough boundary (4, L): boundary condition of VOF function F = 0: undefined = 1: fixed value = 2: Free = 5: wave boundary = 7: Radiation boundary	
INDS(NUMI*NUMJ*NUMK)	I*4	The I, J, K coordinates of the surface cell (I + NUMI * (J - 1) + NUMI * NUMJ * (K - 1))	Figure 0-4-11
INDBK(NUMB)	I*4	Boundary condition of turbulent energy = -2: Gradient zero (do not evaluate advection term) = -1: Value fixed (no advection term is evaluated) = 0: undefined = 1: Fixed value (evaluate advection term) = 2: Evaluate gradient zero advection term) = 6: logarithmic law = 8: Perfect rough boundary	Figure 0-4-9(2)
INDBE(NUMB)	I*4	Boundary conditions of turbulent energy dissipation = -2: Gradient zero (do not evaluate advection term) = -1: Value fixed (no advection term is evaluated) = 0: undefined = 1: Fixed value (evaluate advection term) = 2: Evaluate gradient zero advection	Figure 0-4-9(2)

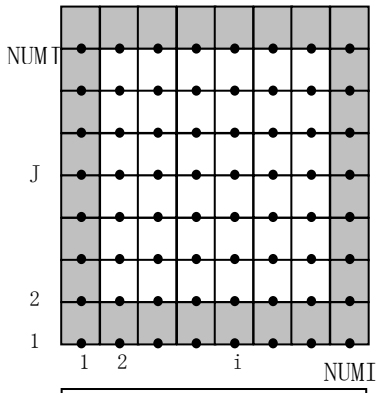
		term) = 6: logarithmic law = 8: Perfect rough boundary	
INDBT(NUMB)	I*4	Temperature boundary condition = -4: heat transfer (no advection term is evaluated) = -3: Heat flux (no advection term is evaluated) = -2: Thermal insulation (no advection terms evaluated) = -1: Temperature fixed (no advection term is evaluated) = 0: undefined = 1: Temperature fixed (evaluate advection term) = 2: Thermal insulation (evaluate advection term) = 3: Heat flux (Evaluating advection term) = 4: heat transfer (evaluate advection term)	Figure 0-4-9(2)
INDBC(NUMB, LEQC)	I*4	Concentration boundary condition = -4: Mass transfer (do not evaluate advection term) = -3: diffusion flux (no advection term is evaluated) = -2: Gradient zero (do not evaluate advection term) = -1: Concentration fixed (Advective term not evaluated) = 0: undefined = 1: Concentration fixed (evaluate advection term) = 2: Gradient zero (evaluate advection term)	Figure 0-4-9(2)

		= 3: diffusion flux (to evaluate advection term) = 4: Mass transfer (evaluate advection term)	
IPVC(NUMI, NUMJ, NUMK)	I*4	Index for calculation of air pressure	
		= 0: non-bubble cell	
		> = 1: bubble number	
IBUF(NUMBUF*MAXBUF)	I*4	Buffer for parallel	—
NWK1(NUMI, NUMJ, NUMK)	I*4	Work array	—
NWKBC(NUMB)	I*4	Work array	—




```
      DO 120 K=2,NUMK-1
        DO 110 J=2,NUMK-1
          DO 100
            I=3,NUMI-1
            ..............
100         CONTINUE
110       CONTINUE
120     CONTINUE
```

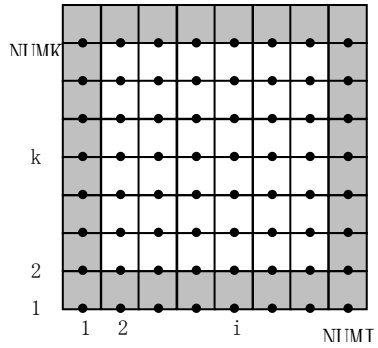
(1) Array defined on x plane




```
      DO 120 K=2,NUMK-1
        DO 110 J=3,NUMK-1
          DO 100
            I=2,NUMI-1
            ..............
100         CONTINUE
111       CONTINUE
120     CONTINUE
```

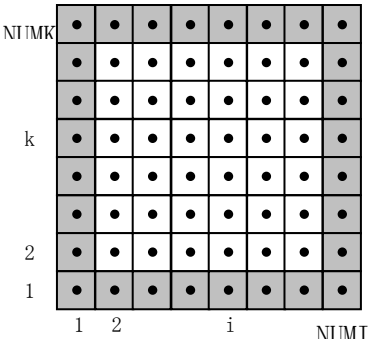
(2) Array defined on y plane

Figure 0-4-6 Array defined on XY plane




```
      DO 120 K=3,NUMK-1
        DO 110 J=2,NUMK-1
          DO 100
            I=2,NUMI-1
            ..............
100         CONTINUE
112       CONTINUE
```

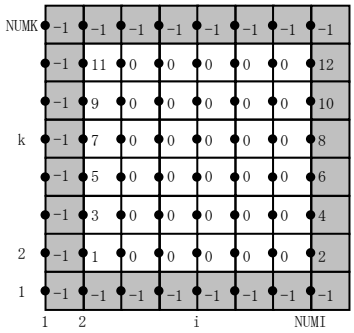
(1) Array defined on z plane



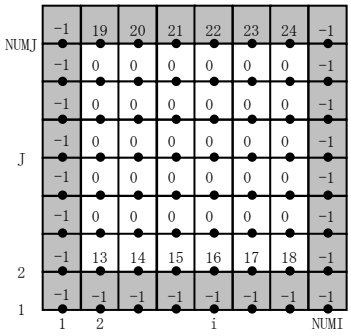

```
      DO 120 K=2,NUMK-1
        DO 110 J=2,NUMK-1
          DO 100
            I=2,NUMI-1
            ..............
100         CONTINUE
113       CONTINUE
```

(2) Array defined on cell center

Figure 0-4-7 Array defined on z plane and cell center

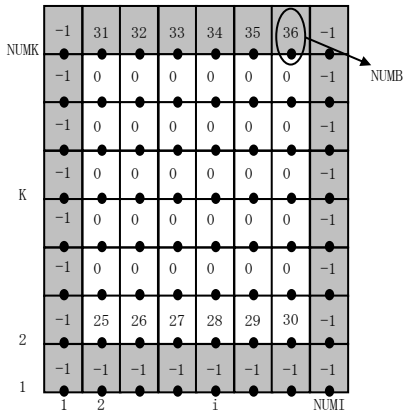


(1) INDX

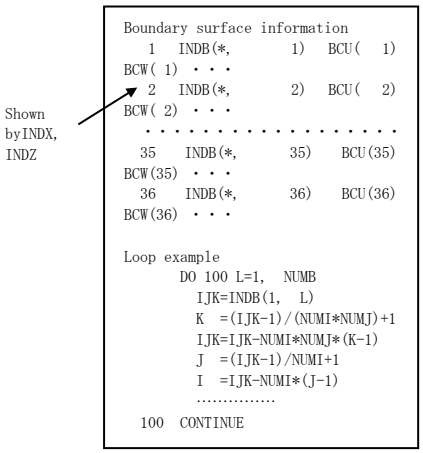


(2) INDY

Figure 0-4-8 INDX, INDY



(1) INDZ



(2) INDB, Boundary value

Figure 0-4-9 INDZ, INDB, Boundary value

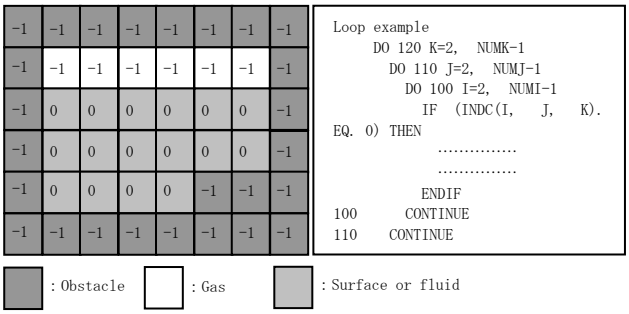


Figure 0-4-10 INDC

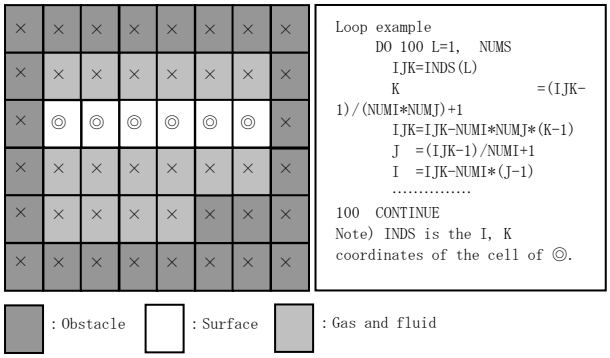


Figure 0-4-11 INDS

Chapter 5 **Use of program**

In this section, we show how to use CADMAS-SURF/3D-MG programs such as restrictions, input / output files and drawing part.

5.1. Restrictions

Since CADMAS-SURF/3D-MG was developed based on FORTRAN 90, it corresponds to the dynamic allocation of arrays, so there is no restriction on the number of grids and the number of boundary surfaces (as far as the memory allows) , The user needs to change the upper limit or the like of the number of data and recompile as necessary.

●MAXNPI：Maximum number of processes in x direction (for parallel use)

This variable is described in the parameter statement in the file “VF_A0PRM.h”. All compilation is done when changing, default is 10.

●MAXNPJ：Maximum number of processes in y direction (for parallel use)

This variable is described in the parameter statement in the file “VF_A0PRM.h”. All compilation is done when changing, default is 10.

●MAXCHR：Maximum number of characters per line (input file)

This variable is described in the parameter statement in the file VF_A0PRM. H All compilation is done when changing, default is 256.

●MAXWDS：Maximum number of words per line (input file)

This variable is described in the parameter statement in the file VF_A0PRM. H All compilation is done when changing, default is 128.

●MAXNC：Maximum number of components of concentration

This variable is described in the parameter statement in the file “VF_A0PRM.h”. All compilation is done when changing, default is 10.

●MAXTR：Maximum number of output target data to time series file

This variable is described in the parameter statement in the file ““VF_A0PRM.h”.”. All compilation is done when changing, default is 10000.

●MAXPRB：Maximum number of spatial blocks of time dependent porosity

This variable is described in the parameter statement in the file ““VF_A0PRM.h”.”. All compilation is done when changing, default is 10.

●MAXPVC：Maximum number of bubbles (for calculating air pressure)

This variable is described in the parameter statement in the file “VF_A0PRM.h”. All compilation is done when changing, default is 500.

●MAXDR：Maximum number of coefficients of Dupuit-Forheimer expression

This variable is described in the parameter statement in the file “VF_A0PRM.h”. All compilation is done when changing, default is 10.

5.2. Input file

5.2.1. Example of Input file

The input file is a file for inputting analysis conditions and is created using commercially available word processing software or the like. File name is data. In, it is a text file. The input file is also required at the time of restart calculation, there is no restriction except that the initial value of the time step width becomes invalid and the structure / boundary condition should not be changed. Most of the items can be changed. Examples of input files are shown below.

Table 0-5-1 Input file example

Input file	Contents
##############################	comment
# TEST DATA	
##############################	
PARALLEL X 100	Parallel control data
PARALLEL X 200	
PARALLEL X 300	
TIME AUTO 0. 001 0. 20	Time control data
TIME LIMIT 1. 0D-5 0. 05	
TIME END 999999 40. 1	
MATE W-LEVEL 10. 0	Physical property data etc.
MATE DENSITY 1000. 0	
MATE K-VISC 0. 0	
MATE GRAVITY 9. 8	
MATE I. C. V 0. 0 0. 0 0. 0	
MODEL WAVE-BC X- FUNC STREAM 5	Wave model data
MODEL WAVE-BC X- DEPTH 10. 0	

MODEL WAVE-BC X- HEIGHT 4. 0 MODEL WAVE-BC X- PERIOD 8. 007 MODEL WAVE-BC X- AMPL 2. 0	
MODEL DAMP X+ DEGREE 2 MODEL DAMP X+ PARAM-XY 0. 6 MODEL DAMP X+ PARAM-Z 0. 6 MODEL DAMP X+ WIDTH 146. 0804 MODEL DAMP X+ DEPTH 10. 0	Attenuation region data
MODEL OPEN-BC X+ FUNC TYPE1 MODEL OPEN-BC X+ DEPTH 10. 0 MODEL OPEN-BC X+ PERIOD 8. 007	Open boundary data
COMP SCHM VP-DONOR 0. 2 COMP MTRX M-ILUBCGSTAB 0. 95 COMP MTRX MAX-ITR 500 COMP MTRX A-ERROR 1. 0D-12 COMP MTRX R-ERROR 1. 0D-10	Numerical solution related data
GRID X 0. 00000000 0. 91300250 1. 82600500 2. 73900750 3. 65201000 ～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～Omitted ～～ 360. 63598750 361. 54899000 362. 46199250 363. 37499500 364. 28799750 365. 20100000 END GRID Y 0. 00 1. 00 END GRID Z 0. 000 0. 400 0. 800 1. 200 1. 600 ～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～Omitted ～～	Grid coordinate data

18. 000 18. 400 18. 800 19. 200 19. 600 20. 000 END	
B. C. D VP SLIP B. C. D F FREE	Boundary condition data
FILE L/P AREA XZ 1 1 FILE L/P OFF OBST FILE L/P OFF BC-IND FILE L/P OFF CM0 FILE L/P OFF NF FILE L/P OFF F FILE L/P OFF V FILE L/P OFF P FILE L/P OFF VISC FILE L/P OFF BC	File control data
FILE GRP TIME 0. 0 9999. 9 8. 007	
FILE TRN STEP 0 999999 1 FILE TRN W-LEVEL ANS X- FILE TRN W-LEVEL 1 1 FILE TRN W-LEVEL 81 1 FILE TRN W-LEVEL 321 1 FILE TRN W-LEVEL 400 1	
OPTION T-DOOR BUB OFF OPTION T-DOOR DROP OFF	Option data

5.2.2. Input file rules

The input file is composed of lines and is decomposed into words by one or more blanks.The inside of the line is all free format and the following parts are ignored There is no rule of order of input data.

- Blank line
- Continuous blanks from the beginning of the line
- Characters after "#" (for comment)
- Characters after the last word valid in grammar

The maximum number of characters in one line is MAXCHR of file VF_A0PRM. h, and the maximum number of words in one line is MAX WDS of file “VF_A0PRM.h”.

Please note that special characters such as tabs are not judged.

5.2.3. Content of input file

The contents of the input file are shown below, as shown in Figure 0-5-1, do not include virtual cells in the suffix in the input file.

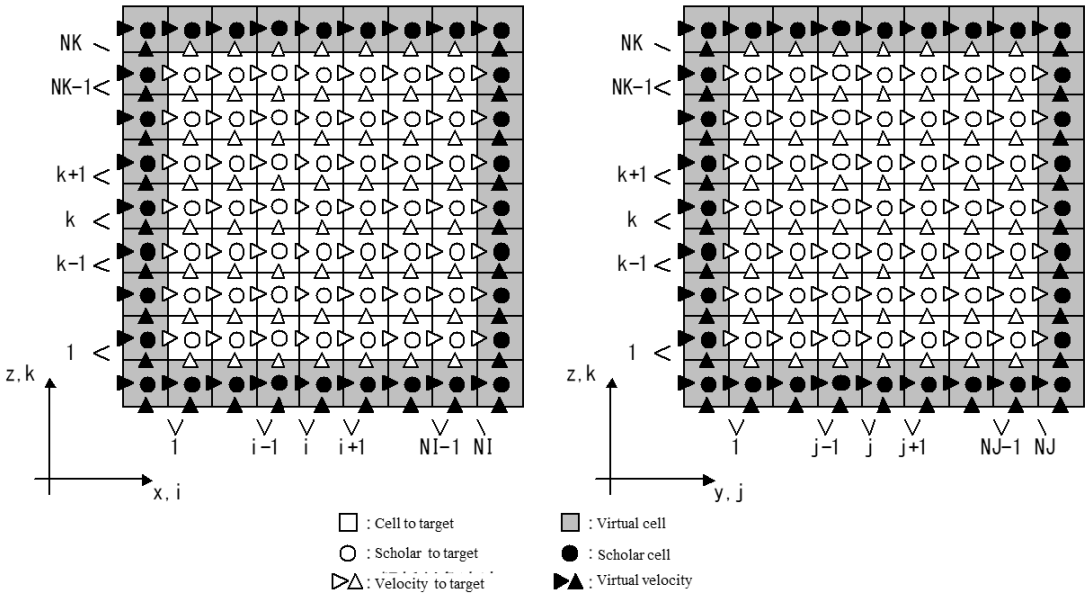


Figure 0-5-1 Suffix in input file (NI, NJ, NK is the number of grid coordinates input)

==========

========== Parallel control ==========

==========

■Parallel control

PARALLEL X [IC]
PARALLEL Y [JC]

Variable name	Description	Type	Default	Restriction
IC	End cell number in x direction	I*4	-	2=<IC=<NI-2
JC	End cell number in y direction	I*4	-	2=<JC=<NJ-2

Note) The number of inputs per direction of this data + 1 is the number of processes by direction

==========

========== Equation control ==========

==========

■Equation control

EQUATION K-EPS {ISW}
EQUATION TEMPERATURE {ISW}
EQUATION CONCENTRATION [NC]

Variable name	Description	Type	Default	Restriction
{ISW}	Selection whether to calculate 'NOCALC' : No 'CALC' : Yes	C*(*)	-	Specified string only
[NC]	Number of components	I*4	0	0=<NC=<MAXNC

Note) Do not calculate by default
When NS = 0, do not calculate

==========

========== Time control ==========

==========

■Time control

TIME CONST [DTCNST]
TIME AUTO [DTINIT] [DTSAFE]
TIME LIMIT [DTMIN] [DTMAX]
TIME END [NEND] [TEND]

Variable name	Description	Type	Default	Restriction
DTCNST	Constant value of time step[s]	R*8	1. 0D-3	DTCNST>=ZERO
DTINIT	Initial value of time step[s]	R*8	1. 0D-6	DTINIT>=ZERO
DTSAFE	Safety rate of time step[-]	R*8	1. 0D-1	DTSAFE>=ZERO
DTMIN	Minimum time step[s]	R*8	ZERO	ZERO=<DTMIN=<DTMAX
DTMAX	Maximum value of time step[s]	R*8	1. 0D0/ZERO	ZERO=<DTMIN=<DTMAX
NEND	Analysis end step [-]	I*4	0	NEND>=0
TEND	Analysis end time [s]	R*8	0. 0D0	TEND>=0. 0D0

Note) The default is "AUTO"
"AUTO" does not consider limitation by wave velocity
The initial value becomes invalid at restart

==========

========== Physical property ==========

==========

■Physical property

< For flow field >
MATE W-LEVEL [WVLVL]
MATE DENSITY [RHO0]
MATE K-VISC [ANU0]
MATE GRAVITY [GRZ0]

MATE I. C.　　V [UINI] [VINI] [WINI]

<k-ε>
MATE I. C.　　KE [AKINI]　[AEINI]

< For temperature field >
MATE S-HEAT　　[CP]
MATE T-COND　　[TC]
MATE T-DENS　　[T0]　[DRDT]
MATE I. C.　　T [TINI]

< For concentration field >
MATE DIFFUS　　[LC]　[DF]
MATE C-DENS　　[LC]　[C0]　[DRDC]
MATE I. C.　　C [LC]　[CINI]

Variable name	Description	Type	Default	Restriction
WVLVL	Initial water surface height [m]	R*8	0. 0D0	-
RHO0	Density [kg/m3]	R*8	1. 0D+3	RHO0>=ZERO
ANU0	Molecular kinematic viscosity coefficient [m2/s]	R*8	1. 0D-6	ANU0>=0. 0
GRZ0	Gravitational acceleration [m/s2]	R*8	9. 8	GZR0>=0. 0
UINI	Initial value of x direction flow velocity [m/s]	R*8	0. 0	-
VINI	Initial value of y direction flow velocity [m/s]	R*8	0. 0	-
WINI	Initial value of z direction flow velocity [m/s]	R*8	0. 0	-
AKINI	Initial value of turbulent energy [m2/s2]	R*8	ZERO	AKINI>=ZERO
AEINI	Initial value of turbulent energy dissipation [m2/s3]	R*8	ZERO	AEINI>=ZERO
CP	Constant pressure specific heat [J/K/kg]	R*8	4. 2D+3	CP>=ZERO
TC	Thermal conductivity [W/m/K]	R*8	5. 7D-1	TC>=ZERO
T0	Reference temperature for buoyancy calculation [K]	R*8	2. 73D+2	-

DRDT	Temperature differential of density [kg/m3/K]	R*8	1. 5D-1	-
TINI	Initial value of temperature [K]	R*8	2. 73D+2	-
LC	Concentration component number	R*8	-	
1=<LS=<NS				
DF	Diffusion coefficient [m2/s]	R*8	-	
DF>=ZERO				
C0	Reference concentration for buoyancy calculation [-]	R*8	0. 0D0	-
DRDC	Concentration derivative of density [kg/m3]	R*8	0. 0D0	-
CINI	Initial value of concentration [K]	R*8	0. 0D0	-

Note) The gravitational acceleration when using the wave form model shall be 9.8

Buoyancy density is calculated from RHO = RHO 0 - DRDT * (T - T 0) - DRDC * (C - C 0)

==========

========== Model ==========

==========

■Wave maker

```
MODEL WAVE-BC    {DIR}   FUNC      STREAM   [N]
MODEL WAVE-BC    {DIR}   FUNC      {TYPE}
MODEL WAVE-BC    {DIR}   DEPTH       [D]
MODEL WAVE-BC    {DIR}   HEIGHT      [H]
MODEL WAVE-BC    {DIR}   PERIOD      [T]
MODEL WAVE-BC    {DIR}   AMPL        [A]
MODEL WAVE-BC    {DIR}   AREA    [LC1]   [LC2]
MODEL WAVE-BC    {DIR}   ANGLE   [ANG]    [X0]    [Y0]
MODEL WAVE-SRC {DIR2}              [LC]
MODEL WAVE-SRC           FUNC      STREAM   [N]
MODEL WAVE-SRC           FUNC      {TYPE}
MODEL WAVE-SRC           DEPTH       [D]
MODEL WAVE-SRC           HEIGHT      [H]
MODEL WAVE-SRC           PERIOD      [T]
MODEL WAVE-SRC           AMPL        [A]
```

Variable name	Description	Type	Default	Restriction
{DIR}	Position of waveguide boundary and direction of propagation	C*(*)	-	

Specified character only

'X -': From the minimum x coordinate position to the normal direction

'X +': From the maximum x coordinate position to the normal direction

'Y-': From y coordinate minimum position to normal direction

'Y +': From y coordinate maximum position to normal direction

{DIR2} Propagation direction of wave source C*(*) - Specified string only

'X' : To x direction

'Y' : To y direction

N Order of flow function method B I*4 - 1=<N=<22

{TYPE} Waveform functions other than flow function method B C*(*) -

Specified string only

'STK-CND' : Stokes wave or quanoidal wave

'STOKES' : Stokes wave

'CNOIDAL' : Quanoidal wave

'MATRIX' : Matrix data -1

'MATRIX2' : Matrix data -2 (only BC)

D Depth[m] R*8 - D>=ZERO

H Wave depth[m] R*8 - H>=ZERO

T Periods[s] R*8 - T>=ZERO

A Cycles to amplify R*8 0. 0 -

LC1 If {DIR} is 'Y+'or'Y-'

x direction cell number (starting point) I*4 1 1=<LC1=<LC2=<NI-1

If {DIR} is 'X+'or'X-'

y direction cell number (starting point) I*4 1 1=<LC1=<LC2=<NJ-1

LC2 If {DIR} is 'Y+'or'Y-'

x direction cell number (end point) I*4 NI-1 1=<LC1=<LC2=<NI-1

If {DIR} is 'X+'or'X-'

y direction cell number (end point) I*4 NJ-1 1=<LC1=<LC2=<NJ-1

LC If {DIR2} is 'X'

x direction cell number I*4 - 1=<LC=<NI-1

If {DIR2} is 'Y'

y direction cell number I*4 - 1=<LC=<NJ-1

ANG A counterclockwise angle with the normal direction of each boundary plane taken as zero [DEG] R*8 0. 0 -

X0 Reference point for phase calculation (x coordinate[m]) R*8 - -

Y0	Reference point for phase calculation (y coordinate [m])	R*8	-	-

Note) Default of wave function is 'STK - CND'

When using wave function other than matrix data, water depth, wave height and period are essential

For matrix data the following is mandatory

Water level / flow rate: depth and period are essential

Flow velocity: period is essential

Water level: period is essential (unusable for wave generating source)

In the case of A <ZERO, a predetermined wave is generated from the beginning

The default for AREA is the whole area

ANG can not be used for matrix data

ANG defaults to zero

Reference point is not used when | ANG | = <ZERO

■ Open boundary condition

MODEL OPEN-BC {DIR} FUNC {TYPE}

MODEL OPEN-BC {DIR} DEPTH [D]

MODEL OPEN-BC {DIR} PERIOD [T]

Variable name	Description	Type	Default	Restriction
{DIR}	Position of open boundary and direction of propagation	C*(*)	-	Specified string only
	'X -': From the minimum x coordinate position to the normal direction			
	'X +': From the maximum x coordinate position to the normal direction			
	'Y-': From y coordinate minimum position to normal direction			
	'Y +': From y coordinate maximum position to normal direction			
{TYPE}	Type of open boundary	C*(*)	-	Specified string only
	'TYPE1' : Radiation boundary due to micro amplitude waves			
D	Depth[m]	R*8	-	D>=ZERO
T	Period[s]	R*8	-	T>=ZERO

Note) When TYPE 1 is used, water depth and cycle are required

■Damping area

MODEL DAMP {DIR} DEGREE [N]
MODEL DAMP {DIR} PARAM-XY [PXY]
MODEL DAMP {DIR} PARAM-Z [PZ]
MODEL DAMP {DIR} WIDTH [W]
MODEL DAMP {DIR} DEPTH [D]

Variable name	Description	Type	Default	Restriction
{DIR}	Position of Damping area 'X -': Near the x coordinate minimum position 'X +': Near the x coordinate maximum position 'Y -': Near y coordinate minimum position 'Y +': Near the y coordinate maximum position	C*(*)	-	Specified string only
N	Degree of Damping function [-]	I*4	-	N>=0
PXY	Horizontal direction parameter of attenuation function [-]	R*8	0. 6	PXY>=ZERO
PZ	Vertical parameter of damping function [-]	R*8	0. 6	PZ >=ZERO
W	Width of the damping region [m]	R*8	-	W >=ZERO
D	Depth of damping [m]	R*8	-	D >=ZERO

Note) When attenuation region is used, degree, width and water depth are essential
The recommended value of the order of the attenuation function is 2
The default value of the attenuation function is the recommended value when the width of the region is two wavelengths.

■Turbulent model

MODEL K-EPS LIM [AKMIN] [AEMIN]
MODEL K-EPS PARAMETER [CMU] [SGK] [SGE] [C1] [C2] [C3]
MODEL K-EPS LOG-LAW [K0] [A0]
MODEL K-EPS PRANDTL [PR]
MODEL K-EPS SCHMIDT [LC] [SM]

Variable name	Description	Type	Default	Restriction
AKMIN	Minimum value of turbulent energy [m2/s2]	R*8	ZERO	>=ZERO

AEMIN	Minimum value of turbulent energy dissipation [m2/s3]	R*8	ZERO	>=ZERO
CMU	Cμ[-]	R*8	0. 09	>=ZERO
SGK	σk[-]	R*8	1. 0	>=ZERO
SGE	σe[-]	R*8	1. 3	>=ZERO
C1	C1[-]	R*8	1. 44	>=0. 0
C2	C2[-]	R*8	1. 92	>=0. 0
C3	C3[-]	R*8	0. 0	>=0. 0
K0	κ[-]	R*8	0. 4	>=ZERO
A0	A[-]	R*8	5. 5	>=ZERO
PR	Turbulent Prandtl number [-]	R*8	1. 0D0	PR>=ZERO
LC	Concentration component number	R*8	-	1=<LS=<NS
SM	Turbulent Schmidt number [-]	R*8	1. 0D0	SM>=ZERO

==========

========== Numerical solution ==========

==========

■Numerical solution

COMP SCHM VP-DONOR [SCMVP]
COMP SCHM FF-DN-AC
COMP SCHM FF-SLOPE
COMP SCHM KE-DONOR [SCMK]
COMP SCHM T-DONOR [SCMT]
COMP SCHM C-DONOR [LC] [SCMC]
COMP MTRX ILUBCGSTAB
COMP MTRX M-ILUBCGSTAB [CGPARA]
COMP MTRX MAX-ITR [ICGMAX]
COMP MTRX A-ERROR [CGEPSA]
COMP MTRX R-ERROR [CGEPSR]

Variable name	Description	Type	Default	Restriction
SCMVP	Difference scheme parameter of advection term	R*8	1. 0	0. 0=<SCMVP=<1. 0
SCMK	Difference scheme parameter of advection term	R*8	1. 0	0. 0=<SCMK=<1. 0

SCMT Difference scheme parameter of advection term R*8 1. 0 0. 0=<SCMT=<1. 0

LC Concentration component number R*8 - 1=<LS=<NS

SCMC Difference scheme parameter of advection term R*8 1. 0 0. 0=<SCMC=<1. 0

CGPARA Parameters for MILU [-] R*8 0. 95 0. 0=<CGPARA=<1. 0

ICGMAT Maximum number of iterations [-] I*4 500 ICGMAX>=0

CGEPSA Convergence determination value (absolute error) [-] R*8 1. 0D-15 CGEPSA>=0. 0

CGEPSR Convergence determination value (relative error)[-] R*8 1. 0D-12 CGEPSR>=0. 0

Note : The default for the difference scheme is DONOR

"ILUBCGSTAB" or "M - ILUBCGSTAB" can be selected, and the default is "M - ILUBCGSTAB"

==========

========== Grid coordinate ==========

==========

■Grid coordinate (Required)

GRID {COORD} [S(1)] [S(2)] . . . . .

. . . . . .

. . . . . [S(N)] END

Variable name	Description	Type	Default	Restriction
{COORD}	Type of coordinates 'X' : x coordinate 'Y' : y coordinate 'Z' : z coordinate	C*(*)	-	Specified string only
S(i)	Coordinate number[m]	R*8	-	S(i)-S(i-1)>=ZEROG
N	Grid number	I*4	-	N>=2

Note) Input must be 1 degree

The number of coordinate values input per line is arbitrary

N (NI, NJ or NK) is not an explicit input, the total number of coordinate values input

==========

==========Obstacle ==========

==========

■Obstacle (File input)

OBST FILE {FILE}

Variable name	Description	Type	Default	Restriction
{FILE}	External file name	C*(*)	-	-

When this option is specified, OBST data is read from an external file.

External file is created by cutting and pasting a line starting with OBST of existing data

■Obstacle (Rectangle input)

OBST [IC1] [JC1] [KC1] [IC2] [JC2] [KC2]

Variable name	Description	Type	Default	Restriction
IC1	X direction cell number (starting point of rectangle)	I*4	-	1=<IC1=<IC2=<NI-1
JC1	Y direction cell number (start point of rectangle)	I*4	-	1=<JC1=<JC2=<NJ-1
KC1	Z direction cell number (start point of rectangle)	I*4	-	1=<KC1=<KC2=<NK-1
IC2	X direction cell number (end point of rectangle)	I*4	-	1=<IC1=<IC2=<NI-1
JC2	Y direction cell number (end point of rectangle)	I*4	-	1=<JC1=<JC2=<NJ-1
KC2	Z direction cell number (end point of rectangle)	I*4	-	1=<KC1=<KC2=<NK-1

==========

========== Porosity ==========

■Porosity value

POROUS FILE {FILE}

Variable name	Description	Type	Default	Restriction
{FILE}	External file name	C*(*)	-	-

When this option is specified, POROUS data is read from an external file.
External file is created by cutting and pasting a line starting with POROUS of existing data.

■The lower limit value data of the porous value

POROUS LIM [PL]

■Porosity

POROUS V [IC1] [JC1] [KC1] [IC2] [JC2] [KC2] [V1] [V2]

■X-direction area transmittance

POROUS X [IG1] [JC1] [KC1] [IG2] [JC2] [KC2] [V1] [V2]

■Y-direction area transmittance

POROUS Y [IC1] [JG1] [KC1] [IC2] [JG2] [KC2] [V1] [V2]

■Z-direction area transmittance

POROUS Z [IC1] [JC1] [KG1] [IC2] [JC2] [KG2] [V1] [V2]

■Inertia force coefficient

POROUS CM [IC1] [JC1] [KC1] [IC2] [JC2] [KC2] [CM]

■Resistance coefficient

POROUS CD [IC1] [JC1] [KC1] [IC2] [JC2] [KC2] [CD]

Variable name	Description	Type	Default	Restriction
PL	Lower limit of the porous values [-]	R * 8	1 0 D - 4	ZERO = <PL = <1. 0
IG1	x direction grid number (starting point of rectangle)	I * 4	- 1	= <IG 1 = <IG 2 = <NI
JG1	y direction grid number (starting point of rectangle)	I * 4	- 1	= <JG 1 = <JG 2 = <NJ
KG1	z direction grid number (starting point of rectangle)	I * 4	- 1	= <KG 1 = <KG 2 = <NK
IG2	x direction grid number (end point of rectangle)	I * 4	- 1	= <IG 1 = <IG 2 = <NI
JG2	y direction grid number (end point of rectangle)	I * 4	- 1	= <JG 1 = <JG 2 = <NJ
KG2	z direction grid number (end point of rectangle)	I * 4	- 1	= <KG 1 = <KG 2 = <NK
IC1	x direction cell number (starting point of rectangle)	I * 4	- 1	= <IC 1 = <IC 2 = <NI - 1
JC1	y-direction cell number (starting point of rectangle)	I * 4	- 1	= <JC 1 = <JC 2 = <NJ - 1
KC1	z-direction cell number (starting point of rectangle)	I * 4	- 1	= <KC 1 = <KC 2 = <NK - 1
IC2	x direction cell number (end point of rectangle)	I * 4	- 1	= <IC 1 = <IC 2 = <NI - 1
JC2	y direction cell number (end point of rectangle)	I * 4	- 1	= <JC 1 = <JC 2 = <NJ - 1
KC2	z-direction cell number (end point of rectangle)	I * 4	- 1	= <KC 1 = <KC 2 = <NK - 1
V1	Ratio of water in V1 structure [-]	R * 8	- 0. 0	= <V 1 = <1.0.
V2	Ratio of water according to V2 shape [-]	R * 8	- 0. 0	= <V 2 = <1.0.
CM	inertial force coefficient [-]	R * 8	-	CM> = 0. 0
CD	resistance coefficient [-]	R * 8	-	CD> = 0.0

Note) When a porous value less than PL is input, replace with PL

Set to a part other than obstacles of the specified rectangle (can be set on the surface of the obstacle)

PL = <Water proportion = V2 + (1.0 - V2) * V1 = <1.0.

==========

========== Boundary condition ==========

==========

■Boundary condition (Required)

B. C. D {BC}

B. C. X [IG1] [JC1] [KC1] [IG2] [JC2] [KC2] {BC}

B. C. Y [IC1] [JG1] [KC1] [IC2] [JG2] [KC2] {BC}

B. C. Z [IC1] [JC1] [KG1] [IC2] [JC2] [KG2] {BC}

Variable name	Description	Type	Default	Restriction
IG1	x direction grid number (starting point of rectangle)	I*4	-	1=<IG1=<IG2=<NI
JG1	y direction grid number (starting point of rectangle)	I*4	-	1=<JG1=<JG2=<NJ
KG1	z direction grid number (starting point of rectangle)	I*4	-	1=<KG1=<KG2=<NK
IG2	x direction grid number (end point of rectangle)	I*4	-	1=<IG1=<IG2=<NI
JG2	y direction grid number (end point of rectangle)	I*4	-	1=<JG1=<JG2=<NJ
KG2	z direction grid number (end point of rectangle)	I*4	-	1=<KG1=<KG2=<NK
IC1	x direction cell number (starting point of rectangle)	I*4	-	1=<IC1=<IC2=<NI-1
JC1	y-direction cell number (starting point of rectangle)	I*4	-	1=<JC1=<JC2=<NJ-1
KC1	z-direction cell number (starting point of rectangle)	I*4	-	1= 1=<KC1=<KC2=<NK-1
IC2	x direction cell number (end point of rectangle)	I*4	-	1=<IC1=<IC2=<NI-1
JC2	y-direction cell number (end point of rectangle)	I*4	-	1=<JC1=<JC2=<NJ-1
KC2	z-direction cell number (end point of rectangle)	I*4	-	1=<KC1=<KC2=<NK-1
{BC}	Type of boundary condition	C*(*)	-	Specified string and value combination

velocity and pressure boundary condition

Slip : VP SLIP

Non slip : VP NON-S

Fixed Velocoty : VP FIX-V [BU] [BV] [BW]

Free : VP FREE

Logarithmic law : VP LOG

Rough surface : VP LOG-KS [KS]

Boundary condition of VOF function F

Slip: F FIX [BF]

Slip: F FREE

Boundary condition of turbulent energy

Value fixed (-A): K FIX-A [BK]

Value fixed (+ A): K FIX + A [BK]

Gradient zero (-A): K FREE-A

Gradient zero (+ A): K FREE + A

Boundary conditions of turbulent energy dissipation

Value fixed (-A): E FIX-A [BE]

Value fixed (+ A): E FIX + A [BE]

Gradient zero (-A): E FREE-A

Gradient zero (+ A): E FREE + A

Temperature boundary condition

Temperature fixed (-A): T FIX-A [BT]

Temperature fixed (+ A): T FIX + A [BT]

Insulation (-A): T FREE-A

Adiabatic (+ A): T FREE + A

Heat flux (-A): T FLUX-A [BTQ]

Heat flux (+ A): T FLUX + A [BTQ]

Heat transfer (-A): T TRAN-A [BTH] [BT 0]

Heat transfer (+ A): T TRAN + A [BTH] [BT 0]

Concentration boundary condition

Concentration fixed (-A): C [LC] FIX-A [BC]

Concentration fixed (+ A): C [LC] FIX + A [BC]

Gradient zero (-A): C [LC] FREE-A

Gradient zero (+ A): C [LC] FREE + A

Diffusion flux (-A): C [LC] FLUX-A [BCQ]

Diffusion flux (+ A): C [LC] FLUX + A [BCQ]

Mass transfer (-A): C [LC] TRAN-A [BCH] [BC 0]

Mass transfer (+ A): C [LC] TRAN + A [BCH] [BC 0]

BU	x direction flow velocity value [m/s]	R*8	-	-
BV	y direction flow velocity value [m/s]	R*8	-	-
BW	z direction flow velocity value [m/s]	R*8	-	-
KS	Wall roughness [m]	R*8	-	>=ZERO

BF	Value of VOF function F [-]	R*8	-	-
BK	turbulence energy [m 2 / s 2]	R * 8	-	> = ZERO
BE	Turbulent energy dissipation [m 2 / s 3]	R * 8	-	> = ZERO
BT	temperature [K]	R * 8	-	-
BTQ	heat flux [W / m 2]	R * 8	-	-
BTH	heat transfer coefficient [W / m 2 / K]	R * 8	-	BTH> = ZERO
BT0	External temperature [K]	R * 8	-	-
LC	concentration component number	R * 8	-	1 = <LS = <NS
BC	concentration [-]	R * 8	-	-
BCQ	diffusion flux [m / s]	R * 8	-	-
BCH	mass transfer coefficient [m / s]	R * 8	-	BCH> = ZERO
BC0	External concentration [-]	R * 8	-	-

Note) Set to the boundary surface within the specified rectangle.

Waveform boundaries and open boundaries are prioritized when separately specifying wave boundary and open boundary.

In the plane where the logarithmic law and the complete rough surface are specified for the flow velocity, the same boundary is also set for the turbulent flow rate.

[+ A] flows the scalar quantity of the boundary when evaluating the advection term (inflow other than the specified flux etc.).

[-A] does not flow boundary scalar quantity when evaluating advection term.

==========

==========File control ==========

==========

■Setting of reading topographical data at restart

FILE OBS {FLAG}

Variable name	Description	Type	Default	Restriction
FLAG	Selection to use topographical data at restart	C*(*)	OFF	OFF または ON

Note) When this option is set to ON, topographic data of the responsible area of each PE at the initial calculation

(OBST, POROUS) to the obs file, and when restart calculation obs file

Read, ignore OBST and POROUS headers.

In order to achieve high speed using this option, it is necessary to designate OBST FILE and POROUS FILE and set the OBST data and POROUS data to external files, otherwise the OBST or POROUS header This is because reading time can not be reduced.
If you do not convert it to an external file, the number of OBST and POROUS header reads is 3, it becomes 1 when it is converted to an external file, and when FILE RES AUTO is set, The number of times becomes zero.

■Restart file control data

FILE RES [ISTEP]
FILE RES AUTO

Variable name	Description	Type	Default	Restriction
ISTEP	Step to restart [-]	I*4	-	ISTEP>=0

Note) When restarting, the initial value of the time step width becomes invalid.
Changing the position of the structure is impossible (not checked in the program).
Change of boundary condition is impossible (program is not checked).
If it is set to AUTO, read the value of ISTEP from the ars file.
If reading from the ars file can not be performed, initial calculation is performed instead of restart calculation It is preferable to use the function of AUTO in combination with specification of FILE RSL ELAPSE.

■List file output control data

FILE L/P STEP [ISTR] [IEND] [ISTEP]
FILE L/P TIME [TSTR] [TEND] [TSTEP]
FILE L/P AREA {SECT} [IJKC] [IJKG]
FILE L/P ON {PHYS}
FILE L/P OFF {PHYS}

Variable name	Description	Type	Default	Restriction
ISTR	Output start step [-]	I*4	-	ISTR=<IEND
IEND	Output end step [-]	I*4	-	ISTR=<IEND
ISTEP	Output step interval [-]	I*4	-	ISTEP>0
TSTR	Output start time[s]	R*8	-	TSTR=<TEND

TEND	Output end time[s]	R*8	-	TSTR=<TEND
TSTEP	Output time interval [s]	R*8	-	TSTEP>0. 0
{SECT}	Output section	C*(*)	'XY'	Specified string only

'XY' : xy section

'XZ' : xz section

'YZ' : yz section

'YX' : yx section

'ZX' : zx section

'ZY' : zy section

IJKC	Cell number of output section	I*4	0	0=<IJKC=<N-1
IJKG	Grid number of output section	I*4	0	0=<IJKG=<N
{PHYS}	Specified output physical quantity	C*(*)	-	Specified string only

'OBST': Obstacle

'BC-IND': Index on boundary condition

'CM 0': Inertial force coefficient

'CD 0': resistance coefficient

'POROUS': Porous value

'LAMBDA': GLV, GLX, GLZ

'NF': NF

'F': VOF function F

'V': flow velocity

'P': Pressure

'K': turbulent energy

'E': Turbulent energy dissipation

'VISC-T': Vortical viscosity coefficient

'VISC': sum of molecular dynamic viscosity and vortical viscosity coefficient

'T': temperature

'T-COND': Thermal conductivity

'C': Concentration

'DIFFUS': diffusion coefficient

'BC': boundary value

'IPVC': index for calculation of air pressure

'PPPVC': Air pressure (S cell also air pressure)

Note) Output at the interval specified by STEP or TIME (not output by default).

When IJKC = 0, the physical quantity defined at the cell center is output with 0 = <IJKC = <N.

When IJKG = 0, the physical quantities defined at the cell interface center are output as 0 = <IJKG = <N.

By default all quantities are non-output .

■Graphic file output control data

FILE GRP STEP [ISTR] [IEND] [ISTEP]
FILE GRP TIME [TSTR] [TEND] [TSTEP]
FILE GRP AREA [IC1] [JC1] [KC1] [IC2] [JC2] [KC]
FILE GRP ON VORT
FILE GRP OFF VORT

Variable name	Description	Type	Default	Restriction
ISTR	Output start step[-]	I*4	-	ISTR=<IEND
IEND	Output end step[-]	I*4	-	ISTR=<IEND
ISTEP	Output step interval[-]	I*4	-	ISTEP>0
TSTR	Output start time[s]	R*8	-	TSTR=<TEND
TEND	Output end time[s]	R*8	-	TSTR=<TEND
TSTEP	Output time interval[s]	R*8	-	TSTEP>0. 0
IC1	Cell number in x direction of start point[-]	I*4	1	1=<IC1=<IC2<NI-1
JC1	Cell number in y direction of start point[-]	I*4	1	1=<JC1=<JC2<NJ-1
KC1	Cell number in z direction of start point [-]	I*4	1	1=<KC1=<KC2<NK-1
IC2	Cell number in x direction of end point [-]	I*4	NI-1	1=<IC1=<IC2<NI-1
JC2	Cell number in y direction of end point [-]	I*4	NJ-1	1=<JC1=<JC2<NJ-1
KC2	Cell number in z direction of end point [-]	I*4	NK-1	1=<KC1=<KC2<NK-1

Note) Output at intervals specified by STEP or TIME (not output by default)

Output area is specified by AREA (all areas by default)

A region of at least 2 × 2 cells is necessary (the direction of 1 cell is not shown)
If [* c1 = <0], it is interpreted as [* c1 = 1] and [* c2 <= 0] as [* c2 = N-1]
The vorticity is substituted by three components of density component [+1] to [+3] in the drawing part. Vorticity is not output by default.

■Detailed file output control data

FILE RSL STEP [ISTR] [IEND] [ISTEP]
FILE RSL TIME [TSTR] [TEND] [TSTEP]
FILE RSL ELAPSE [ETIME]

Variable name	Description	Type	Default	Restriction
ISTR	Output start step[-]	I*4	-	ISTR=<IEND
IEND	Output end step[-]	I*4	-	ISTR=<IEND
ISTEP	Output step interval[-]	I*4	-	ISTEP>0
TSTR	Output start time[s]	R*8	-	TSTR=<TEND
TEND	Output end time[s]	R*8	-	TSTR=<TEND
TSTEP	Output time interval[s]	R*8	-	TSTEP>0. 0
ETIME	Elapsed time since start of calculation (s)	R*8	-	ETIME>0. 0

Note) Output at intervals specified by STEP or TIME (not output by default)
ELAPSE may overlap the designation of TIME or STEP.
If ELAPSE is specified, it is checked whether or not the elapsed time of each step exceeds ETIME in the time integration loop. If it exceeds ETIME, the restart file and the ars file are output and the calculation is terminated.
The function of ELAPSE is preferably used in combination with the specification of FILE RES AUTO.

■Time series file output control data

FILE TRN STEP [ISTR] [IEND] [ISTEP]
FILE TRN TIME [TSTR] [TEND] [TSTEP]
FILE TRN W-LEVEL ANS {DIR}
FILE TRN W-LEVEL [IC1] [JC1]
FILE TRN FORCE {FDIR} [IC1] [JC1] [KC1] [IC2] [JC2] [KC2]

FILE TRN POINT {PHY} [I1] [J1] [K1]
FILE TRN POINT VORT-X [IC1] [JC1] [KC1]
FILE TRN POINT VORT-Y [IC1] [JC1] [KC1]
FILE TRN POINT VORT-Z [IC1] [JC1] [KC1]
FILE TRN {CALC} {PHY} [IC1] [JC1] [KC1] [IC2] [JC2] [KC2]

Variable name	Description	Type	Default	Restriction
ISTR	Output start step[-]	I*4	-	ISTR=<IEND
IEND	Output end step[-]	I*4	-	ISTR=<IEND
ISTEP	Output step interval[-]	I*4	-	ISTEP>0
TSTR	Output start time[s]	R*8	-	TSTR=<TEND
TEND	Output end time[s]	R*8	-	TSTR=<TEND
TSTEP	Output time interval[s]	*8	-	TSTEP>0. 0
I1	Grid or Cell number in x direction	I*4	-	Cell number : 1=<I1=<NI-1
	(grid number only for U)			Grid number : 1=<I1=<NI
J1	Grid or Cell number in y direction	I*4	-	Cell number : 1=<J1=<NJ-1
	(grid number only for V)			Grid number : 1=<J1=<NJ
K1	Grid or Cell number in z direction	I*4	-	Cell number : 1=<K1=<NK-1
	(grid number only for W)			Grid number : 1=<K1=<NK
IC1	Cell number in x direction	I*4	-	1=<IC=<NI-1
JC1	Cell number in y direction	I*4	-	1=<JC=<NJ-1
KC1	Cell number in z direction	I*4	-	1=<KC=<NK-1
IC2	Cell number in x direction	I*4	-	1=<IC=<NI-1
JC2	Cell number in y direction	I*4	-	1=<JC=<NJ-1
KC2	Cell number in z direction	I*4	-	1=<KC=<NK-1
{DIR}	Position of wave boundary and direction of propagation	C*(*)	-	Specified strings only

'X -': Solving the wave boundary at the x coordinate minimal position
'X +': Solving the wave boundary at the x coordinate maximum position
'Y -': Solving the wave boundary at y coordinate minimum position
'Y +': Solving the wave boundary at the y coordinate maximum position

'SRC': solution of the wave source

{FDIR} Direction of force C*(*) - Specified strings only

'X-': Wave force in the negative direction of x to an obstacle

'X +': Wave force in the positive direction of x to an obstacle

'Y -': Wave force in the negative direction of y to obstacles

'Y +': positive wave force of y to the obstacle

'Z-': Wave force in the negative direction of z to an obstacle

'Z +': positive wave force of z to the obstacle

{PHY} Physical quantity to output C * (*) - Specified character string only

'U': x direction flow velocity

'V': Y direction flow velocity

'W': flow velocity in z direction

'P': Pressure

'F': F value

'K': turbulent energy

'E': Turbulent energy dissipation

'T': temperature

'C' [LC]: concentration (LC is below)

{CALC} Calculation method C * (*) - Specified character string only

'MIN': Minimum value

'MAX': Maximum value

'AV': Volume average value

'INT': volume integrated value

LC concentration component number R * 8 - 1 = <LS = <NS

Note) The meaning of the vorticity value is as follows

VORT - X: @ w / @ y - @ v / @ z

VORT - Y: @ u / @ z - @ w / @ x

VORT - Z: @ v / @ x - @ u / @ y

The minimum value in parallel calculation is restricted up to 1.0 D + 30

The maximum value in parallel calculation is -1. Limited to 0 D + 30

■Time dependent porosity file control data

FILE PORO [NT]

Variable name	Description	Type	Default	Restriction
NT	Number of data of the porosity in the time direction	I*4	0	>=0

Note) When NT = 0, do not read porosity file
When NT = 1, it is not time dependent data (it can be used as a file reading function of porosity)

■エージェントモデル用出力制御データ

FILE MAM STEP [ISTR] [IEND] [ISTEP]
FILE MAM TIME [TSTR] [TEND] [TSTEP]

Variable name	Description	Type	Default	Restriction
ISTR	Output start step[-]	I*4	-	ISTR=<IEND
IEND	Output end step[-]	I*4	-	ISTR=<IEND
ISTEP	Output step interval[-]	I*4	-	ISTEP>0
TSTR	Output start time[s]	R*8	-	TSTR=<TEND
TEND	Output end time[s]	R*8	-	TSTR=<TEND
TSTEP	Output time interval[s]	R*8	-	TSTEP>0. 0

Note) Output at intervals specified by STEP or TIME (not output by default)

==========

========== オプションデータ ==========

==========

■Processing related to air bubbles is not performed

OPTION T-DOOR BUB OFF

■Raising bubbles (Timer door method)

OPTION T-DOOR BUB [WBUB]

Variable name	Description	Type	Default	Restriction
WBUB	Rising speed of bubble [m/s]	R*8	0. 2	WBUB>=ZERO

Note) By default, it is increased by 0.2 [m / s]

■Processing on water drops is not performed (Timer door method)

OPTION T-DOOR DROP OFF

■Processing on water drops is performed (Timer door method)

OPTION T-DOOR DROP FREE-RUNDOWN

■Calculation of the flow velocity of the surface cell with zero slope

OPTION S-CELL-VEL DU=0

■Extrapolation of the flow velocity of the surface cell

OPTION S-CELL-VEL D2U=0

■Sub-loop data of flow velocity / pressure calculation

OPTION SUB-LOOP [LOOP]

Variable name	Description	Type	Default	Restriction
LOOP	Number of sub loops	I*4	1	LOOP>=1

Note) When the sub loop is once, it is the same as normal calculation

■Calculation of air pressure

OPTION PV=CONST [PVCP0] [PVCGM]

Variable name	Description	Type	Default	Restriction

PVCP0 Atmospheric pressure [Pa] R*8 -
PVCP0>=ZERO
PVCGM Specific heat ratio [-] R*8 -
PVCGM>=1. 0

Note) Pressure value output by L / P output etc is subtracted from atmospheric pressure
The air bubbles contacting the wave formation, radiation, free are set to atmospheric pressure

■ Resistance force calculation by Dupuit-Forheimer equation

OPTION DRAG-DF [YU] [N]
[DR1] [AP1] [BT1]
[DR2] [AP2] [BT2]
. . . . . .
[DRi] [APi] [BTi]
. . . . . .
[DRN] [APN] [BTN]

Variable name	Description	Type	Default	Restriction
YU	Kinematic viscosity coefficient [m2/s]	R*8	-	>0. 0
N	Number of input of coefficient	I*4	-	>0
DRi	Rubble of rubble [m]	R*8	-	>0. 0
APi	coefficientα0	R*8	-	>=0. 0
BTi	coefficientβ0	R*8	-	>=0. 0

When 0 0 <DR <= DR 1 α 0 = 0. 0, β 0 = 0. 0
When DR 1 <DR ≤ DR 2 α 0 = AP 1, β 0 = BT 1
When DR 2 <DR ≤ DR 3 α 0 = AP 2, β 0 = BT 2 . . . . . .
When DRN<DR, α0=APN, β0=BTN

* Comment lines and blank lines are not allowed in [OPTION DRAG-DF] lines or less [N] lines
* Be careful as the meaning of existing CD 0 is the grain size [m] of riprap
* Dynamic viscosity coefficient is entered separately from "MATE K - VISC"
* Currently N = <10 (MAXDRG = 10 of VF_A 0 PRM. H)

■Do not set water level change amount data

OPTION SEA-BOTTOM OFF

■Set water level variation data (read sbt file)

OPTION SEA-BOTTOM ON

■Calculate water level variation data from tomographic parameters (fault_cad. Txt file reading)

OPTION SEA-BOTTOM CALC

Note) In addition, it is necessary to specify the coordinate system of calculation grid (OPTION COORDINATE) as well.
Also, if necessary, specify the datum system specification (OPTION FAULT - SYSTEM, OPTION GRID - SYSTEM)

■Specification of the coordinate system of the calculation grid(in the case of the plane rectangular coordinate system)

OPTION COORDINATE JAPAN-PLANE-RECTANGULAR [ICOORD]

Variable name	Description	Type	Default	Restriction
ICOORD	Region number of plane rectangular coordinate system	I*4	-	1<=ICOORD<=19

Note) Valid only for OPTION SEA - BOTTOM CALC

■Specification of the coordinate system of calculation grid (for UTM coordinate system)

OPTION COORDINATE UTM [LC_DEG]

Variable name	Description	Type	Default	Restriction
LC_DEG	Longitude of central meridian of UTM coordinate system (degree)	R*8	-	0<=LC_DEG<360

Note) Valid only for OPTION SEA - BOTTOM CALC

■Specification of the datum of the fault parameter

OPTION FAULT-SYSTEM {SYS}

Variable name	Description	Type	Default	Restriction
SYS	Name of geodetic system	C*(*)	JGD2000	TOKYO, JGD2000, or WGS84

Note) Valid only for OPTION SEA - BOTTOM CALC

■Designation of geodetic system of calculation grid

OPTION GRID-SYSTEM {SYS}

Variable name	Description	Type	Default	Restriction
SYS	Name of geodetic system	C*(*)	JGD2000	TOKYO, JGD2000, or WGS84

Note) Valid only for OPTION SEA - BOTTOM CALC

■Specification of flow rate limit value

OPTION MAX-VELOCITY [VVMAX]

Variable name	Description	Type	Default	Restriction
VVMAX	Flow rate limit	R*8	-	0<VVMAX

Note) The limit value is applied after calculating the temporary flow velocity

==========

========== Data for debugging==========

==========

■F value specification data to rectangular box

DEBUG F-BOX [IC1] [JC1] [KC1] [IC2] [JC2] [KC2] [FF]

Variable name	Description	Type	Default	Restriction
IC1	Cell number in x direction (Start point of the rectangle)	I*4	-	1=<IC1=<IC2=<NI-1
JC1	Cell number in y direction (Start point of the rectangle)	I*4	-	1=<JC1=<JC2=<NJ-1
KC1	Cell number in z direction (Start point of the rectangle)	I*4	-	1=<KC1=<KC2=<NK-1
IC2	Cell number in x direction (End point of the rectangle))	I*4	-	1=<IC1=<IC2=<NI-1
JC2	Cell number in y direction (End point of the rectangle)	I*4	-	1=<JC1=<JC2=<NJ-1
KC2	Cell number in z direction (End point of the rectangle)	I*4	-	1=<KC1=<KC2=<NK-1
FF	F value	R*8	-	0=<FF=<1. 0

Note) Only one box entered last

■Speed specification data for TD to rectangular box

DEBUG TD-VEL [IC1] [JC1] [KC1] [IC2] [JC2] [KC2] [U] [V] [W]

Variable name	Description	Type	Default	Restriction
IC1	Cell number in x direction (Start point of the rectangle)	I*4	-	1=<IC1=<IC2=<NI-1
JC1	Cell number in y direction (Start point of the rectangle)	I*4	-	1=<JC1=<JC2=<NJ-1
KC1	Cell number in z direction (Start point of the rectangle)	I*4	-	1=<KC1=<KC2=<NK-1
IC2	Cell number in x direction (End point of the rectangle)	I*4	-	1=<IC1=<IC2=<NI-1
JC2	Cell number in y direction (End point of the rectangle)	I*4	-	1=<JC1=<JC2=<NJ-1
KC2	Cell number in z direction (End point of the rectangle)	I*4	-	1=<KC1=<KC2=<NK-1

U	x direction velocity	R*8	-	-
V	y direction velocity	R*8	-	-
W	z direction velocity	R*8	-	-

Note) Only one box entered last

5.3. List file

List file is a file that outputs the physical time for each analysis step, the number of iterations of the BiCGSTAB method, etc., and displays it using commercially available word processor software etc. File name is data. List and it is a text format file. An example of a list file is shown below.

List file	Description
##### CADMAS-SURF/3D-MG Ver. 1. 0 START. ######	Execution start comment
##### MYRANK= 0 / 4	Rank and number of processors
##### DEFAULT.	Default setting
##### INPUT-DATA.	Read input file
## INPUT-LEVEL= 0 [PARALLEL][X][100] [PARALLEL][X][200] [PARALLEL][X][300] [GRID][X] & [0. 00000000][0. 91300250][1. 82600500][2. 73900750][3. 65201000] ～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～ Omitted～～ & [18. 000][18. 400][18. 800][19. 200][19. 600][20. 000] & [END]	Input level 0 (determination of the number of grids etc.) Echo of parallel control data Echo of grid data
## INPUT-LEVEL= 1	Input level 1 (determination of the number of boundary surfaces)

## INPUT-LEVEL= 2	Input level 2 (input of various data)
[TIME][AUTO][0. 001][0. 20]	Echo of various data
[TIME][LIMIT][1. 0D-5][0. 05]	
～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～	
Omitted～～	
[OPTION][T-DOOR][DROP][OFF]	Construction of various information
##### SETUP.	Output analysis condition
##### CONDITION.	
-- PARALLEL --	Parallel control data
NPROCS [-] = 4	
NPI, NPJ [-] = 4 1	
MYRANK [-] = 0	
MYI, MYJ [-] = 1 1	
CELL-X(ALL) [-] = 400	
ICS, ICE [-] = 1 100	
CELL-Y(ALL) [-] = 1	
JCS, JCE [-] = 1 1	
CELL-Z(ALL) [-] = 50	
KCS, KCE [-] = 1 50	
-- EQUATION --	Equation control data
K-EPS = NOCALC	
TEMPERATURE = NOCALC	
CONCENTRATION = NOCALC	
-- TIME CONTROL --	Time control data
DT TYPE = AUTO	
SAFE [-] = 2. 00000E-01	
INIT [S] = 1. 00000E-03	
MIN [S] = 1. 00000E-05	
MAX [S] = 5. 00000E-02	
END STEP [-] = 999999	
TIME [S] = 4. 01000E+01	

-- MATERIAL --	Physical property data etc.
W-LEVEL [M] = 1. 00000E+01	
DENSITY [KG/M3] = 1. 00000E+03	
K-VISC [M2/S] = 0. 00000E+00	
GRAVITY [M/S2] = 9. 80000E+00	
I. C. U [M/S] = 0. 00000E+00	
V [M/S] = 0. 00000E+00	
W [M/S] = 0. 00000E+00	
-- MODEL --	Model etc. Data
WAVE-BC X-	Wave model data
FUNC = STREAM	
N[-]= 5	
DEPTH [M]= 1. 00000E+01	
HEIGHT [M]= 4. 00000E+00	
PERIOD [S]= 8. 00700E+00	
AMPL [-]= 2. 00000E+00	
LENGTH [M]= 7. 30402E+01	
URSELL NUMB[-]= 2. 51319E+01	
0. 0=WAVE(x)[-]= 7. 97690E-01	
OPEN-BC X+	Open boundary data
FUNC = TYPE1	
DEPTH [M]= 1. 00000E+01	
PERIOD [S]= 8. 00700E+00	
LENGTH [M]= 7. 09303E+01	
VELOCITY [-]= 8. 85853E+00	
DAMP X+	Attenuation region data
DEGREE [-]= 2	
PARAM-XY [-]= 6. 00000E-01	
PARAM-Z [-]= 6. 00000E-01	
WIDTH [M]= 1. 46080E+02	
DEPTH [M]= 1. 00000E+01	
-- COMPUTATION --	Numerical solution related data
SCHM VP-DNR[-] = 2. 00000E-01	

SCHM FF-DN-AC MTRX TYPE = MILU-BCGSTAB PARAM [-] = 9. 50000E-01 I-MAX [-] = 500 A-ERR [-] = 1. 00000E-12 R-ERR [-] = 1. 00000E-10	
-- FILE CONTROL -- RES = NOT READ L/P TYPE = NOT WRITE GRP TYPE = BY TIME START [S] = 0. 00000E+00 END [S] = 9. 99990E+03 DELTA [S] = 8. 00700E+00 AREA1 [-] = 1 1 1 AREA2 [-] = 400 1 50 RSL TYPE = NOT WRITE TRN TYPE = BY STEP START [-] = 0 END [-] = 999999 DELTA [-] = 1 PORO TYPE = NOT READ	File control data
-- OPTION -- SUB-LOOP [-] = 1 S-CELL-VEL = D2U=0 T-DOOR BUB = NOT USE T-DOOR DROP = NOT USE	Option data
-- GRID DATA -- NUMBER OF GRID-X = 102 I X DX CX 0 -9. 13003E-01 9. 13003E-01 0. 00000E+00 ～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～ Omitted～～ 51 2. 00000E+01 4. 00000E-01 4. 00000E-01	Grid coordinate data

-- OBSTACLE --	Obstacle data
-- BOUNDARY --	Boundary condition data
##### INITIAL.	Setting initial condition
>> FILE-GRP : OUT : INITIAL	Output grid number etc. to graphic file
##### CALCULATION.	Start calculation
STEP= 0 : TIME= 0. 00000E+00 : DT = 0. 00000E+00 ～～～ Omitted～～	Step information
: !B! = 0. 00000E+00 : !R! = 0. 00000E+00 : ITR = 0	
>> FILE-GRP : OUT : STEP= 0 : TIME= 0. 00000E+00	Output calculation result to graphic file
～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～ Omitted～～	
STEP= 918 : TIME= 4. 00203E+01 : DT = 4. 00600E-02 ～～～ Omitted～～	Step information
: !B! = 6. 66105E-01 : !R! = 3. 53342E-11 : ITR = 46	
>> FILE-GRP : OUT : STEP= 918 : TIME= 4. 00203E+01	Output calculation result to graphic file
STEP= 919 : TIME= 4. 00614E+01 : DT = 4. 11181E-02 ～～～ Omitted～～	Step information
: !B! = 6. 38653E-01 : !R! = 5. 91756E-11 : ITR = 47	
STEP= 920 : TIME= 4. 01016E+01 : DT = 4. 02112E-02 ～～～ Omitted～～	Step information
: !B! = 7. 00378E-01 : !R! = 3. 73344E-11 : ITR = 44	
##### CPU TIME [S] ##########################	CPU time output
## <<FLOW>>	
## TOTAL 117. 65	
## +-- PRE PROCESS 0. 10	
## +-- CALCULATION 117. 55	
##	+-- FILE I/O 1. 25	
##	+-- VELO & PRES 105. 42	
##		+-- CONV & VISC 4. 08	
##		+-- GENERATION 1. 26	

## | | +-- INTEGRATION 2. 27	
## | | +-- POISSON COEF 2. 53	
## | | +-- POISSON SOLV 82. 84	
## | | +-- V & P MODIF 1. 33	
## | | +-- E. T. C. 11. 11	
## | +-- TEMPERATURE 0. 00	
## | +-- CONCENTRATION 0. 00	
## | +-- K-EPSIRON 0. 00	
## | +-- VOF FUNCTION 9. 06	
## | | +-- CONVECTION 1. 84	
## | | +-- INTEGRATION 0. 50	
## | | +-- MODIF & CUT 0. 85	
## | | +-- NF & T-DOOR 1. 14	
## | | +-- E. T. C. 4. 74	
## | +-- E. T. C. 1. 82	
## +-- E. T. C. 0. 00	
## <<ROUTINE>>	
## +-- VF_P***** 36. 32	
## +-- VF_M1BCGS 82. 77	
## +-- VF_FDROPF 0. 00	
##### NORMAL END. ##########################	End message (normal termination)

Note) Step information

STEP: calculation step

TIME: Physical time

DT: time step width (= tn - tn - 1)

FSUM: sum of (VOF function F value × cell area)

The sum of FCUT: (cutoff VOF function F value × cell area)

! VD!: The square of the (divergence × cell area)

! B!: Squared norm of the right-hand vector of simultaneous linear equations

! R!: The square of the final residual vector of simultaneous linear equations

ITR: Number of repetitions of solving simultaneous linear equations

5.4. Time series file

Time series file is a file that outputs time dependent data and is displayed using commercially available spreadsheet software etc. File name is data. Tran and it is a file in text form. Here is an example of a time series file .

Time series file	Description
1 W-LEVEL ANS X- -1 -1 -1 -1 -1 2 W-LEVEL ------ 1 1 -1 -1 -1 -1 3 W-LEVEL ------ 81 1 -1 -1 -1 -1 4 W-LEVEL ------ 161 1 -1 -1 -1 -1 5 W-LEVEL ------ 241 1 -1 -1 -1 -1 6 W-LEVEL ------ 321 1 -1 -1 -1 -1 7 W-LEVEL ------ 400 1 -1 -1 -1 -1	Output number, output item, grid (cell) number
TIME 1 2 3 ～ Omitted	Output number
0. 00000000E+00 0. 00000000E+00 0. 00000000E+00 0. 00000000E+00 ～Omitted 1. 20000000E-03 -2. 46385964E-11 0. 00000000E+00 0. 00000000E+00 ～ Omitted 2. 64000000E-03 -2. 62285111E-10 0. 00000000E+00 0. 00000000E+00 ～ Omitted 4. 36800000E-03 -1. 18715773E-09 0. 00000000E+00 0. 00000000E+00 ～ Omitted ～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～ Omitted 3. 99802361E+01 8. 27479959E-02 5. 85501252E-02 4. 41974665E-02 ～ Omitted 4. 00202958E+01 2. 19142791E-02 -1. 57292475E-02 -1. 99720589E-02 ～ Omitted 4. 00614144E+01 -3. 88083644E-02 -8. 98002349E-02 -8. 41081049E-02 ～ Omitted 4. 01016249E+01 -9. 65151936E-02 -1. 60249489E-01 -1. 45746660E-01 ～ Omitted	Physical time, calculated value 1, calculated value 2 ...

5.5. Calculation information file

The calculation information file is a file for the user to grasp the progress of the calculation and is output to the console etc. (standard output) Therefore, the file name is a text format file not specifically designated. An example of a calculation information file is shown.

calculation information	Description
##### CADMAS-SURF/3D-MG Ver. 1. 0 START. ######	Execution start comment
##### DEFAULT.	Default setting
##### INPUT-DATA.	Read input file
##### SETUP.	Construction of various information
##### CONDITION.	Output analysis condition
##### INITIAL.	Setting initial condition
##### CALCULATION.	Start calculation
STEP= 0 : TIME= 0. 00000E+00 : DT = 0. 00000E+00	Step information
STEP= 1 : TIME= 1. 20000E-03 : DT = 1. 20000E-03	
STEP= 2 : TIME= 2. 64000E-03 : DT = 1. 44000E-03	
STEP= 3 : TIME= 4. 36800E-03 : DT = 1. 72800E-03	
～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～～	
Omitted	
STEP= 919 : TIME= 4. 00614E+01 : DT = 4. 11186E-02	
STEP= 920 : TIME= 4. 01016E+01 : DT = 4. 02105E-02	
##### NORMAL END. ###########################	End message (normal termination)

5.5. Matrix data file

Matrix data file is a file for inputting the water level, flow velocity, etc. for wave generation of an arbitrary waveform, and creates it using commercially available word processing software etc. File name is data. Mtb, and it is a text format file Here is the format of the matrix data file.

Water level and flow rate are given	When only flow rate is given	When only water level is given
LEVEL-ON [nz] [nt] [T0] [Z1] [Z2]・・・[Znz] T [T1] [e1] [u1] [w1] [u2] [w2] ・ ・ [unz] [wnz] T [T2] [e2] ・	LEVEL-OFF [nz] [nt] [T0] [Z1] [Z2]・・・[Znz] T [T1] [u1] [w1] [u2] [w2] ・ ・ [unz] [wnz] T [T2] ・	LEVEL-ONLY [nt] [T0] T [T1] [e1] T [T2] [e2] ・ ・ ・ ・ ・ ・ ・

[nz] number of data in the vertical direction, [nt]: number of data in the phase direction, [T 0]: initial dimensionless phase, [U *]: x direction flow velocity, [w *]: a vertical position with zero mean water level, [T *]: dimensionless phase, [e *]: water level with zero mean water level, z direction flow velocity, delimiter T: start of data of a certain phase and used for checking the number of data in the program.

Note that the dimensionless phase [T *] is 0. 0 ≤ [T *] ≤ 1.0, and the dimensionless phase decreases with the progression.In the program, in the phase direction and the spatial direction Since various physical quantities are being interpolated, it is not necessary for the grid coordinates and time increments specified by the input data to coincide with [Z *] and [T *] in the matrix data, and in the phase direction When data from t 1 to t 2 is input and the program tries to refer to the value of t outside the range

· If t <t1, use the value of t1
· If t> t2, use the value of t2

(Similarly in the spatial direction).

As an example, the data when the water level is given is shown below.

```
LEVEL-ON        31                    101      0. 796345000
   -1. 00000E+01    -9. 50000E+00    -9. 00000E+00    -8. 50000E+00    -8. 00000E+00
   -7. 50000E+00    -7. 00000E+00    -6. 50000E+00    -6. 00000E+00    -5. 50000E+00
     ～～～～～～～～～～～～～
     2. 50000E+00      3. 00000E+00      3. 50000E+00      4. 00000E+00      4. 50000E+00
     5. 00000E+00
T      0. 00000E+00    2. 57121E+00
     1. 42074E+00    0. 00000E+00
     1. 42190E+00    0. 00000E+00
     ～～～～～～～～～～～～～
     4. 12557E+00    0. 00000E+00
```

5.6. Porosity file

The porosity file is a file for inputting the porosity data at the time specified by the user, which the user separately creates File name is data. Poro and it is a binary format file.

The format of the porosity file is shown below.

(1) Overview of porosity file

Initial block
First time block
Second time block
・・・・・・・・
*m*th time block

(2) Details of initial block

Record	Format	Variable name	Type	Description
1	NB, ISW1, ISW2, ISW3, ISW4	NB	I*4	Number of rectangular areas
		ISW1	I*4	Flag for future extension (0 in the meantime)
		ISW2	I*4	Flag for future extension (0 in the meantime)
		ISW3	I*4	Flag for future extension (0 in the meantime)
		ISW4	I*4	Flag for future extension (0 in the meantime)
2-(NB+1)	I1(IB), J1(IB), K1(IB), I2(IB), J2(IB), K2(IB)	I1(IB)	I*4	The start point (cell number) in the x direction of the IB th rectangular region
		J1(IB)	I*4	The start point (cell number) in the y direction of the IB th rectangular region
		K1(IB)	I*4	The start point (cell number) in the z direction of the IB th rectangular region
		I2(IB)	I*4	The end point (cell number) in the x direction of the IB th rectangular region
		J2(IB)	I*4	The end point (cell number) in the y direction of the IB th rectangular region
		K2(IB)	I*4	The end point (cell number) in the z direction of the IB th rectangular region

(3) Details of the *m* th time block

Record	Format	Variable name	Type	Description
1	TIME, JSW1, JSW2, JSW3, JSW4	TIME	R*8	Times
		JSW1	I*4	Flag for future extension (0 in the meantime)
		JSW2	I*4	Flag for future extension (0 in the meantime)
		JSW3	I*4	Flag for future extension (0 in the meantime)
		JSW4	I*4	Flag for future extension (0 in the meantime)
2-(NB+1)	((GGV(I, J, K), I=I1(IB), I2(IB)), J=J1(IB), J2(IB)), K=K1(IB), K2(IB)))	GGV(I, J, K)	R*4	Porosity at the position of cell numbers I, J, K

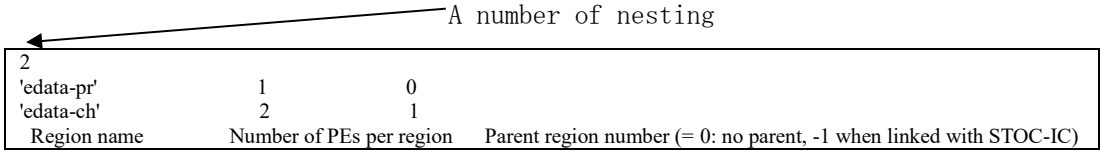
（4） Interpolation in time direction and area transmittance

Linear interpolation is used for temporal interpolation. When the time included in the file is set as [t1 = <t = <t2], the value of t1 is required when the value at the time before t1 is required, and when the value at the time later than t2 is required Use the value of t2. The area transmittance is not entered to prevent the file from becoming large, and the smaller value of the porosity of the cell sandwiching the face is adopted. In the future, when the file input becomes necessary, the flag for extension will be used and improves the program.

.

5.7. MG environment file

The MG environment file is a file for inputting conditions for nesting and using multiple analysis areas of CADMAS-SURF/3D-MG, which the user separately creates File name is data. Env, text The following is an example of the MG environment file when the area shown in Figure 0-5-2 is assumed.



M When using in the MG environment, the input / output File name for each area is not "data. ***", but "area name" + extension (. In,. List,. Mtb Etc). However, since lists and graphic output files are output for each divided area, File name is "area name" + PE number + extension when the number of PEs is larger than 1. For example, in the above example, It is like edata-ch000000. list.

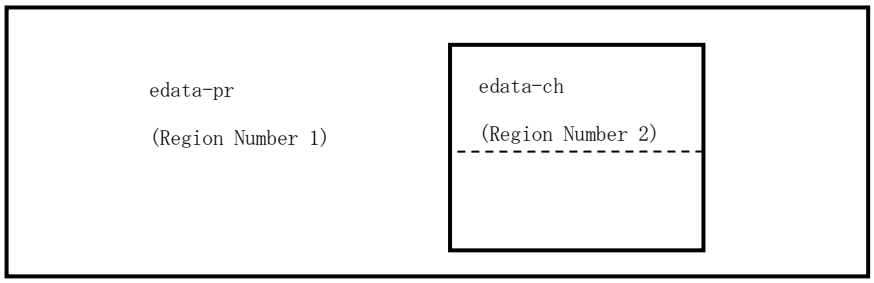


Figure 0-5-2 Assumed area

5.8. Graphic unit

The screens of the Graphic unit are divided into three areas as below.

(1) Command area: Area where the user performs various operations

(2) Main drawing area: Area where analysis result is drawn (drawing ratio can be changed)

(3) Sub Drawing Area: An area for drawing the relation between the analysis area and the drawing area

Two operations, which are not displayed on the screen but can be performed in the main drawing area, are possible.

(1) Control key + mouse drag → enlarge

(2) Shift key + mouse drag → move

As shown in Figure 0-5-4, the physical quantity control command in the command area can be switched by clicking the option button.

When key input is done in the text box, the input is reflected by pressing the "Enter" key at the end.

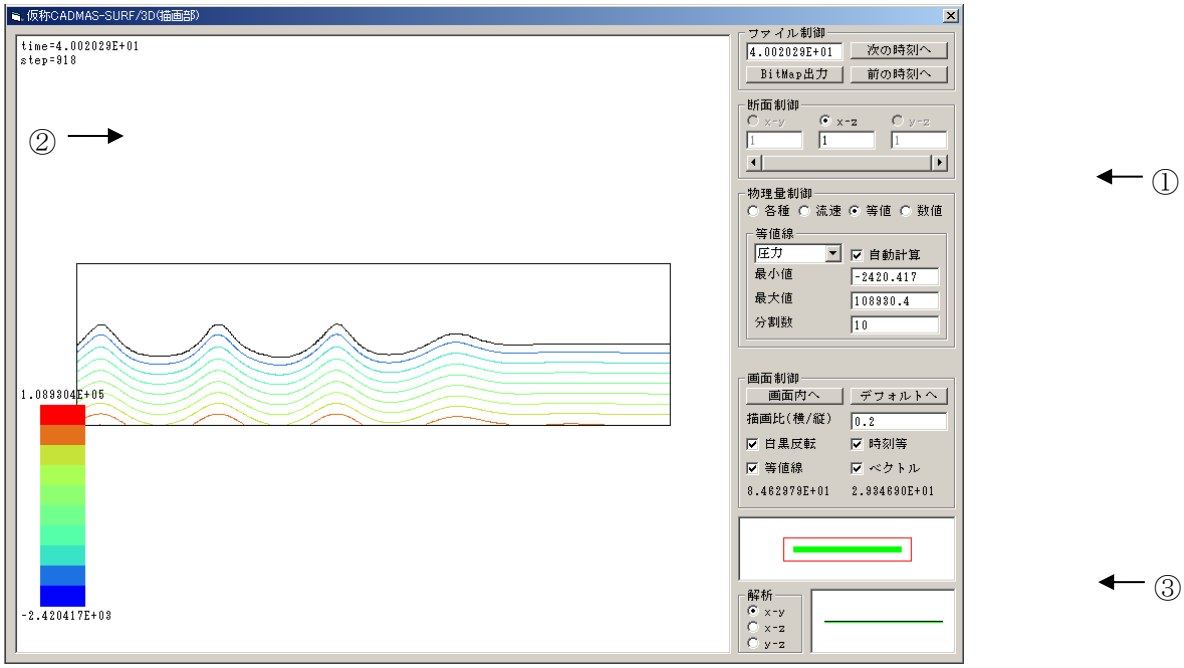


Figure 0-5-3 Image of graphic unit

(a)General (b)Velocity (c) Equivalence (d)Numerical value

Figure 0-5-4 Switching of physical quantity control command

Chapter 6 **Reference**

（1）Coastal Development Institute of Technology. (2001). Research and development of numerical wave channel, Coastal Development of Institute of Technology, CDIT Library, (12)

（2）Sakakiyama, T, Abe, N, Kajima, R. (1990). Analysis of nonlinear wave around permeable structure by porous model, Proceedings of Coastal Engineering, JSCE, 37, 554-558

（3）C. W. Hirt and B. D. Nichols (1981)：Volume of Fluid(VOF) Method for the Dynamics of Free Boundaries, Journal of Computational Physics, Vol. 39, pp. 201-225

（4）M. D. Torrey, R. C. Mjolsness and L. R. Stein (1987)：NASA-VOF3D:A Three-Dimensional Computer Program for Incompressible Flows with Free Surface, LA11009-MS

（5）M. Isobe, H. Nishimura and S. Horikawa: Indication of perturbation solutions of conserved waves by wave heights, Proceedings of the 33rd Annual Conference, II, pp. 760-761, 1978.

（6）Dean, R. G. (1965)：Stream Function Representation of Nonlinear Ocean Waves, J. Geopys. Res. , vol. 70, No. 18, pp. 4561-4572

（7）Iwata, K., Kawasaki, K. and Kim, D. (1994): Numerical Calculation of Wave Breaking by Underwater Structures, Journal of Coastal Engineering, Vol. 42, pp. 781-785.

（8）E.C. Cruz *et al.* Nonreflecting boundary conditions for non-linear wave equation, Proc. Coast. Eng. JSCE(1993)

（9）Numerical Fluid Dynamics Editorial Board (1995): Numerical Fluid Dynamics Series 3 Turbulence Analysis, University of Tokyo

（10）Shimada, A., Nakadai, Y. and Akiyama, M. (1992): Incompressible fluid analysis module I (Cartesian and cylindrical coordinate system), Fuji Research Institute Techniques, Vol. 3 No.

1, pp. 88-99.

（11）Noboru Yamazaki (1992): Incompressible fluid analysis module including free surface, Fuji Research Institute Techniques, Vol. 3 No. 1, pp. 131-146

（12）Takahashi, R. (1982): Computerised Fluid Dynamics (Exercises), Planning Centre.

（13）P. J. Roache(1978)：Computational Fluid Dynamics

（14）Denis Gueyffier, Jie Li, Ali Nadim, Ruben Scardovelli and Stephane Zaleski (1999)：Volume-of –Fluid Interface Tracking with Smoothed Surfase Stress Methods for Tree-Dimensional Flows, Journal of Computational Physics, Vol. 152, pp. 423-456

（15）Fujino, S. (1991): New solution method for asymmetric simultaneous linear equations appearing in thermal-fluid analysis, Proceedings of the 28th Japan Heat Transfer Symposium, II, pp. 622-624