By finite element discretization, the gradient of χ becomes

$$\nabla_Y \chi = [B^e][\chi^e], \tag{34}$$

in a microscopic finite element. $[\chi^e]$ is the derivative of the microscopic displacement for each component of $\overline{\mathbf{F}}$; in other words, it is the matrix given below with nine kinds of characteristic deformation.

$$[\chi^e] = \begin{bmatrix} \chi^e_{111} & \chi^e_{112} & \cdots & \chi^e_{133} \\ \chi^e_{211} & \chi^e_{212} & \cdots & \chi^e_{233} \\ \vdots & \vdots & \ddots & \vdots \\ \chi^e_{n11} & \chi^e_{n12} & \cdots & \chi^e_{n33} \end{bmatrix}, \tag{35}$$

where n is the NDOF of one finite element of the microstructure. The matrix equation becomes

$$\mathbf{K}_\chi [\chi^Q] = [\mathbf{r}_\chi], \tag{36}$$

$$\mathbf{K}_\chi = \int_{Y_0} [B^e]^T [A][B^e] dY, \tag{37}$$

$$[\mathbf{r}_\chi] = \int_{Y_0} [B^e]^T [A][I] dY, \tag{38}$$

from Eq. (33) about one microstructure. $[\chi^Q]$ denotes that the quantity is evaluated at the macroscopic quadrature point. The value of $[\chi^Q]$, which is a (NDOF of the microstructure) row $\times$ 9 column matrix, can be obtained by assembling $[\chi^e]$. $[I]$ is a ninth order identity matrix, consisting of nine column vectors $\{I_i\}(i = 1 \ldots 9)$. Thus nine solutions can be obtained for the right-hand side of Eq. (38), one for each $\{I_i\}$ corresponding to a component of the deformation gradient. $[\chi^Q]$ can then be obtained by solving each different version of the right-hand side of the above equation.

Multiplying by $[\overline{B^e}]$ and dividing by $|V|$ on both sides, Eq. (36) yields

$$\frac{1}{|V|} \mathbf{K}_\chi [\chi^Q][\overline{B^e}] = \frac{1}{|V|} [\mathbf{r}_\chi][\overline{B^e}]. \tag{39}$$

Now, by using Eq. (20)

$$\frac{1}{|V|} \mathbf{K}_\chi = \mathbf{K}^Q_{ww}, \tag{40}$$

$$\frac{1}{|V|} [\mathbf{r}_\chi][\overline{B^e}] = \mathbf{K}^Q_{wu}, \tag{41}$$

and thus,

$$\mathbf{K}^Q_{ww} [\chi^Q][\overline{B^e}] = \mathbf{K}^Q_{wu}. \tag{42}$$

The total is obtained at all macroscopic quadrature points in respect of the above equation

$$\mathbf{K}_{ww}[\chi][\overline{B}] = \mathbf{K}_{wu}, \tag{43}$$

$$[\overline{B}] \equiv \sum_i [\overline{B^e}], \tag{44}$$

$$[\chi] \equiv \sum_i [\chi^Q], \tag{45}$$

where $[\chi]$ is a matrix consisting of (quadrature points of macrostructure $\times$ NDOF of microstructure) rows $\times$ 9 columns.

The characteristic deformations are the deformation increments for unit macroscopic deformation gradients at a particular instant and these describe the material properties and strain distribution of the microstructure. Equation (29) can also be considered a linear approximation of the microscopic deformation. Therefore, the update of the microscopic deformation by

$$\Delta \mathbf{w} = -\chi : \Delta \overline{\mathbf{F}}, \tag{46}$$

corresponds to the Forward Euler method for microscopic deformation from Eq. (32).

2.4 Homogenization method using characteristic deformation mode superposition

In the mode superposition homogenization method, the microscopic displacement increment is approximately obtained by the linear combination of the previously calculated characteristic deformation ${}_0\chi$ and the scaling factor from Eq. (46) as in [10]

$$\Delta w_k \doteqdot -{}_0\chi_{kpQ} \Delta\alpha_{pQ}, \tag{47}$$

where α is the scaling factor for each mode. Inserting Eq. (47) into Eq. (22) yields the matrix

$$\begin{bmatrix} [{}_0\chi]^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{K}_{ww} & \mathbf{K}_{wu} \\ \mathbf{K}_{uw} & \mathbf{K}_{uu} \end{bmatrix} \begin{bmatrix} [{}_0\chi] & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \Delta\alpha \\ \Delta\mathbf{u} \end{Bmatrix} = \begin{bmatrix} [{}_0\chi]^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \mathbf{r}_w \\ \mathbf{r}_u \end{Bmatrix} \tag{48}$$

$$\begin{bmatrix} [{}_0\chi]^T \mathbf{K}_{ww} [{}_0\chi] & [{}_0\chi]^T \mathbf{K}_{wu} \\ \mathbf{K}_{uw} [{}_0\chi] & \mathbf{K}_{uu} \end{bmatrix} \begin{Bmatrix} \Delta\alpha \\ \Delta\mathbf{u} \end{Bmatrix} = \begin{Bmatrix} [{}_0\chi]^T \mathbf{r}_w \\ \mathbf{r}_u \end{Bmatrix}, \tag{49}$$

where $[{}_0\chi]$ is the same kind of matrix as $[\chi]$. The above equation can be represented as

$$\begin{bmatrix} \mathbf{K}_{\alpha\alpha} & \mathbf{K}_{\alpha u} \\ \mathbf{K}_{u\alpha} & \mathbf{K}_{uu} \end{bmatrix} \begin{Bmatrix} \Delta\alpha \\ \Delta\mathbf{u} \end{Bmatrix} = \begin{Bmatrix} \mathbf{r}_\alpha \\ \mathbf{r}_u \end{Bmatrix}, \tag{50}$$

where

$$\mathbf{K}_{\alpha\alpha} = \int_{\Omega} \left(\frac{1}{|V|} \int_{Y_0} [{}_0\chi^e]^T [B^e]^T [A][B^e][{}_0\chi^e] dY \right) dX, \tag{51}$$

$$\mathbf{K}_{\alpha u} = -\int_{\Omega} \left(\frac{1}{|V|} \int_{Y_0} [{}_0\chi^e]^T [B^e]^T [A] dY \right) [\overline{B^e}] dX, \tag{52}$$

$$\mathbf{K}_{u\alpha} = -\int_{\Omega} [\overline{B^e}]^T \left(\frac{1}{|V|} \int_{Y_0} [A][B^e][{}_0\chi^e] dY \right) dX, \tag{53}$$

$$\mathbf{K}_{uu} = \int_{\Omega} [\overline{B^e}]^T \left(\frac{1}{|V|} \int_{Y_0} [A] dY \right) [\overline{B^e}] dX, \tag{54}$$

$$\{\mathbf{r}_u\} = F_{ext} - \int_{\Omega} [\overline{B^e}]^T \left(\frac{1}{|V|} \int_{Y_0} [\Pi] dY \right) dX, \tag{55}$$

$$\{\mathbf{r}_\alpha\} = \int_{\Omega} \left(\frac{1}{|V|} \int_{Y_0} [{}_0\chi^e]^T [B^e]^T [\Pi] dY \right) dX. \tag{56}$$

We obtain the matrix with unknowns $\Delta\alpha$ and $\Delta\mathbf{u}$. Because the NDOF of the matrix is reduced to (NDOF of macrostructure + quadrature point of macrostructure × 9), significant computational cost is saved.

This technique is, however, an approximate means of achieving equilibrium in a range of displacements representing linear combinations of χ_0, as it is clear from Eq. (56) that χ_0 has an effect on the equilibrium. In this way, to approximate the deformation in limited deformation patterns, an approximation error is created depending on the analysis case [15].

3 Algorithm for nonlinear homogenization method

3.1 Generalized algorithm

In Eq. (22), $\{\Delta w\}$ can be statically condensed at the element quadrature point level and becomes

$$\{\Delta\mathbf{w}\} = \mathbf{K}_{ww}^{-1}(\{\mathbf{r}_w\} - \mathbf{K}_{wu}\{\Delta\mathbf{u}\}). \tag{57}$$

$\{\Delta w\}$ vanishes when the above equation is substituted in the macroscopic equilibrium equation

$$(\mathbf{K}_{uu} - \mathbf{K}_{uw}\mathbf{K}_{ww}^{-1}\mathbf{K}_{wu})\{\Delta\mathbf{u}\} = \{\mathbf{r}_u\} - \mathbf{K}_{uw}\mathbf{K}_{ww}^{-1}\{\mathbf{r}_w\}. \tag{58}$$

Now, the microscopic equilibrium hypothesized for $\overline{\mathbf{F}}$ at this time is

$$\{\mathbf{r}_w\} = 0. \tag{59}$$

By using Eqs. (43) and (57), we obtain [5]

$$\{\Delta\mathbf{w}\} = -\mathbf{K}_{ww}^{-1}\mathbf{K}_{wu}\{\Delta\mathbf{u}\} = -[\chi][\overline{\mathbf{B}}]\{\Delta\mathbf{u}\}. \tag{60}$$

By using Eqs. (43), (57), and (58) we can represent

$$(\mathbf{K}_{uu} - \mathbf{K}_{uw}[\chi][\overline{\mathbf{B}}])\{\Delta\mathbf{u}\} = \{\mathbf{r}_u\}, \tag{61}$$

where $(\mathbf{K}_{uu} - \mathbf{K}_{uw}[\chi][\overline{\mathbf{B}}])$ is called the homogenized tangent stiffness.

In differential equation form, from Eqs. (14), (17), and (31), this becomes

$$\int_{\Omega} \delta\overline{\mathbf{F}} : \left(\frac{1}{|V|} \int_{Y_0} \Big(\mathbf{A} : (\mathbf{I} - \nabla_Y \chi) \Big) dY \right) : \Delta\overline{\mathbf{F}} dX$$
$$= \overline{\mathbf{F}}_{ext} - \int_{\Omega} \delta\overline{\mathbf{F}} : \left(\frac{1}{|V|} \int_{Y_0} \Pi dY \right) dX. \tag{62}$$

Microscopic equilibrium, Eq. (59),

$$r_w = \int_{Y_0} \delta\tilde{\mathbf{Z}} : \Pi dY = \mathbf{0}, \tag{63}$$

is a prerequisite of the above equation. To satisfy this nonlinear prerequisite, a Newton–Raphson iteration is needed. Thus the above equation is linearized with respect to $\mathbf{w}$ while $\overline{\mathbf{F}}$ is fixed, and then discretized by the finite element method using Eq. (19). At each quadrature point, the linearized self-equilibrated equation becomes

$$\int_{Y_0} [B^e]^T [A][B^e] dY \{\Delta\mathbf{w}^Q\} = -\int_{Y_0} [B^e]^T [\Pi] dY. \tag{64}$$

In the generalized algorithm, it is necessary to compute three different calculations in each iteration.

1. Update macroscopic tangential homogenization χ using Eq. (36).
2. Solve microscopic equilibrium problem, Eq. (63), and obtain the convergence solution $\{\mathbf{w}\}$ while $\overline{\mathbf{F}}$ is fixed.
3. Solve the linearized macroscopic equilibrium equation, Eq. (61), to obtain $\{\mathbf{u}\}$.

Processes 1 and 2 need to be solved at all quadrature points of the macrostructure and it is known that this contributes the most to the calculation load [6]. These processes require prohibitive computational cost and actual numerical simulation is difficult.

3.2 Block LU factorization algorithm

We now present an algorithm that decreases the residual of each scale simultaneously using the block LU factorization

algorithm without microscopic convergence in the macroscopic iteration as in the conventional algorithm. By block LU factorization, Eq. (22) becomes

$$\begin{bmatrix} \mathbf{K}_{ww} & \mathbf{0} \\ \mathbf{K}_{uw} & \mathbf{S} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{K}_{ww}^{-1}\mathbf{K}_{wu} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \Delta\mathbf{w} \\ \Delta\mathbf{u} \end{Bmatrix} = \begin{Bmatrix} \mathbf{r}_w \\ \mathbf{r}_u \end{Bmatrix}, \tag{65}$$

where $\mathbf{S}$ is called the Schur-Complement. Referring to Eq. (43),

$$\begin{aligned} \mathbf{S} &= \mathbf{K}_{uu} - \mathbf{K}_{uw}\mathbf{K}_{ww}^{-1}\mathbf{K}_{wu} \\ &= \mathbf{K}_{uu} - \mathbf{K}_{uw}[\chi][\overline{\mathbf{B}}] \\ &= \mathbf{K}_{uu} - [\overline{\mathbf{B}}]^T[\chi]^T\mathbf{K}_{ww}[\chi][\overline{\mathbf{B}}], \end{aligned} \tag{66}$$

which corresponds to the homogenized tangent stiffness of Eq. (61). The solution process for Eq. (65) is composed of the forward substitution

$$\begin{bmatrix} \mathbf{K}_{ww} & \mathbf{0} \\ \mathbf{K}_{uw} & \mathbf{S} \end{bmatrix} \begin{Bmatrix} \Delta\tilde{\mathbf{w}} \\ \Delta\tilde{\mathbf{u}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{r}_w \\ \mathbf{r}_u \end{Bmatrix}, \tag{67}$$

and the backward substitution

$$\begin{Bmatrix} \Delta\tilde{\mathbf{w}} \\ \Delta\tilde{\mathbf{u}} \end{Bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{K}_{ww}^{-1}\mathbf{K}_{wu} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \Delta\mathbf{w} \\ \Delta\mathbf{u} \end{Bmatrix}. \tag{68}$$

Then, from Eqs. (43), (67), and (68),

$$\mathbf{K}_{ww}\{\Delta\tilde{\mathbf{w}}\} = \{\mathbf{r}_w\}, \tag{69}$$

$$\mathbf{S}\{\Delta\mathbf{u}\} = \{\mathbf{r}_u\} - \mathbf{K}_{uw}\{\Delta\tilde{\mathbf{w}}\}, \tag{70}$$

$$\begin{aligned} \{\Delta\mathbf{w}\} &= \{\Delta\tilde{\mathbf{w}}\} - \mathbf{K}_{ww}^{-1}\mathbf{K}_{wu}\{\Delta\mathbf{u}\} \\ &= \{\Delta\tilde{\mathbf{w}}\} - [\chi]\Delta\overline{\mathbf{F}}, \end{aligned} \tag{71}$$

hold. The calculation process is described below.

1. Solve linearized microscopic equilibrium equation, Eq. (69), to obtain $\{\Delta\mathbf{w}\}$.
2. Solve linearized macroscopic equilibrium equation, Eq. (70), to obtain $\{\Delta\mathbf{u}\}$. Note that this equation is equivalent to Eq. (58).
3. Update $\{\Delta\mathbf{w}\}$ using Eq. (71). We have already obtained $[\chi]$ from the calculation of $\mathbf{S}$.

Three matrices, namely the micro, macro, and Schur-Complement, need to be solved at each iteration. Although the computational cost of this algorithm is expected to be lower than that of the generalized algorithm since equilibrium of the microstructure is not required at every iteration, the Schur-Complement update is still relatively expensive as described above. A similar algorithm for a differential equation using the Block-Newton method has been proposed by Yamada and Matsui [19].

3.3 Approximation of schur-complement in micro–macro coupled equation

In the block LU factorization algorithm described in the previous subsection, the linearized equations for the microstructure are solved first and then the increment for the macro-displacements is solved using the Schur-Complement that reflects the micro–macro interactions. These two steps essentially define the computational cost because the third step can be solved efficiently by reusing the characteristic deformation that has already been computed in the evaluation of the Schur-Complement in the second step. Although the macroscopic equilibrium conditions must be satisfied as well as the microscopic ones by making a convergence judgment for the residuals, the Schur-Complement is a tangent stiffness for prediction and does not influence the equilibrium directly. Hence, one of the requirements for efficient nonlinear homogenization algorithms is to approximate the Schur-Complement effectively. In this regard, we propose the two algorithms given below.

A simple method is to approximate the Schur-Complement using the previously calculated characteristic deformation ${}_0\chi$ as in

$$\begin{aligned} \mathbf{S} &= \mathbf{K}_{uu} - \mathbf{K}_{uw}[\chi][\overline{\mathbf{B}}] \\ &\doteqdot \mathbf{K}_{uu} - \mathbf{K}_{uw}[{}_0\chi][\overline{\mathbf{B}}] \\ &= \mathbf{K}_{uu} - [\overline{\mathbf{B}}]^T[\chi]^T\mathbf{K}_{ww}[{}_0\chi][\overline{\mathbf{B}}], \end{aligned} \tag{72}$$

that is,

$$[\chi]^T\mathbf{K}_{ww}[\chi] \doteqdot [\chi]^T\mathbf{K}_{ww}[{}_0\chi], \tag{73}$$

is employed from Eq. (66). Henceforth, this method is referred to as the modified block LU factorization algorithm (MBFA). The other approach is to make use of the aforementioned Mode Superposition method in the approximation of the Schur-Complement. The calculation process and an interpretation thereof are given below.

1. Solve the linearized microscopic equilibrium equation, Eq. (69), to obtain $\{\Delta\tilde{\mathbf{w}}\}$.
2. From Eq. (50), solve the mode superposition-based linearized micro–macro coupled equation

$$\begin{bmatrix} \mathbf{K}_{\alpha\alpha} & \mathbf{K}_{\alpha u} \\ \mathbf{K}_{u\alpha} & \mathbf{K}_{uu} \end{bmatrix} \begin{Bmatrix} \Delta\alpha \\ \Delta\mathbf{u} \end{Bmatrix} = \begin{Bmatrix} \mathbf{r}_\alpha \\ \mathbf{r}_u \end{Bmatrix}, \tag{74}$$

   to obtain $\{\Delta\mathbf{u}\}$. Then, $\mathbf{S}$ is approximated by the range of the mode superposition method and the update of $\mathbf{S}$ can be omitted.
3. The assumption of the mode superposition method follows from Eqs. (46), (47), and (71), and $\{\Delta\mathbf{w}\}$ is updated using

$$\{\Delta \mathbf{w}\} = \{\Delta \tilde{\mathbf{w}}\} - [_0\chi]\{\Delta\alpha\}. \tag{75}$$

The meaning of the algorithm is now given. Block LU factorization is applied to Eq. (74) yielding

$$\mathbf{K}_{\alpha\alpha}\{\Delta\tilde{\alpha}\} = \{\mathbf{r}_\alpha\}, \tag{76}$$

$$(\mathbf{K}_{uu} - \mathbf{K}_{u\alpha}\mathbf{K}_{\alpha\alpha}^{-1}\mathbf{K}_{\alpha u})\{\Delta\mathbf{u}\} = \{\mathbf{r}_u\} - \mathbf{K}_{u\alpha}\{\Delta\tilde{\alpha}\}, \tag{77}$$

$$\{\Delta\alpha\} = \{\Delta\tilde{\alpha}\} - \mathbf{K}_{\alpha\alpha}^{-1}\mathbf{K}_{\alpha u}\{\Delta\mathbf{u}\}. \tag{78}$$

In Eq. (77), the macroscopic displacement is updated using

$$(\mathbf{K}_{uu} - \mathbf{K}_{u\alpha}\mathbf{K}_{\alpha\alpha}^{-1}\mathbf{K}_{\alpha u}), \tag{79}$$

which can be considered an approximation of the homogenized tangent stiffness. Then, using Eqs. (43) and (66), the Schur-Complement becomes

$$\begin{aligned}
\mathbf{S} &= \mathbf{K}_{uu} - \mathbf{K}_{uw}[\chi][\overline{\mathbf{B}}] \\
&= \mathbf{K}_{uu} - \mathbf{K}_{uw}[\chi]([\chi]^T\mathbf{K}_{ww}[\chi])^{-1}([\chi]^T\mathbf{K}_{ww}[\chi])[\overline{\mathbf{B}}] \\
&= \mathbf{K}_{uu} - \mathbf{K}_{uw}[\chi]([\chi]^T\mathbf{K}_{ww}[\chi])^{-1}[\chi]^T\mathbf{K}_{wu} \\
&\doteqdot \mathbf{K}_{uu} - \mathbf{K}_{uw}[_0\chi]([_0\chi]^T\mathbf{K}_{ww}[_0\chi])^{-1}[_0\chi]^T\mathbf{K}_{wu} \\
&= \mathbf{K}_{uu} - [\overline{\mathbf{B}}]^T[\chi]^T\mathbf{K}_{ww}[_0\chi] \\
&\quad \times([_0\chi]^T\mathbf{K}_{ww}[_0\chi])^{-1}[_0\chi]^T\mathbf{K}_{ww}[\chi][\overline{\mathbf{B}}] \\
&= \mathbf{K}_{uu} - \mathbf{K}_{u\alpha}\mathbf{K}_{\alpha\alpha}^{-1}\mathbf{K}_{\alpha u},
\end{aligned} \tag{80}$$

that is,

$$\begin{aligned}
&[\chi]^T\mathbf{K}_{ww}[\chi] \doteqdot [\chi]^T\mathbf{K}_{ww}[_0\chi] \\
&\quad \times([_0\chi]^T\mathbf{K}_{ww}[_0\chi])^{-1}[_0\chi]^T\mathbf{K}_{ww}[\chi],
\end{aligned} \tag{81}$$

is employed. In the mode superposition method, since the homogenized tangent stiffness corresponds to the exact value with the mode updated at every Newton–Raphson iteration as described above, this method can give an approximate mean of the homogenized tangent stiffness by decreasing the number of times that the mode is updated. This method is referred to as the mode superposition algorithm (MSA).

4 Numerical examples

4.1 Comparison of computational costs and convergence properties

With respect to computational cost and convergence, which are of interest to us, we now compare the four methods introduced in the previous section, namely, the generalized algorithm (GA), block LU factorization algorithm (BFA), modified block LU factorization algorithm (MBFA) and the algorithm using mode superposition (MSA). Detailed algorithms for the parallel computation of each of the methods are given in Fig. 2. The GA calculates the microscopic equilibrium at each quadrature point in every iteration. In contrast to the GA, the BFA, MBFA, and MSA decrease the residual

1. Set calculation condition
2. Perform in a unit cell at each quadrature point by paraell computation
   Solve microscopic equilibrium problem : $\{\mathbf{r}_w\} = 0$
   Update the microscopic deformation : $\{\mathbf{w}\}$
   Solve for update χ : $\mathbf{K}_\chi[\chi^Q] = [\mathbf{r}_\chi]$
   Compute : $\mathbf{K}_{uu} - \mathbf{K}_{uw}\mathbf{K}_{ww}^{-1}\mathbf{K}_{wu}, \{\mathbf{r}_u\}$
3. Communication using MPIallreduce
   Assemble : $\mathbf{K}_{uu} - \mathbf{K}_{uw}\mathbf{K}_{ww}^{-1}\mathbf{K}_{wu}, \{\mathbf{r}_u\}$
4. Solve
   $$(\mathbf{K}_{uu} - \mathbf{K}_{uw}[\chi][\overline{\mathbf{B}}])\{\Delta\mathbf{u}\} = \{\mathbf{r}_u\}$$
5. Update the vatiables
   $$\{\mathbf{u}\} = \{\mathbf{u}\} + \{\Delta\mathbf{u}\}$$
6. if $|\{\mathbf{r}_u\}| < Tol$, then
   Next step GOTO 1
   else
   Next iteration GOTO 2

(a)

1. Set calculation condition
2. Perform in a unit cell at each quadrature point by paraell computation
   Compute : $\mathbf{K}_{ww}, \{\mathbf{r}_w\}$
   Solve : $\mathbf{K}_{ww}\{\Delta\tilde{\mathbf{w}}\} = \{\mathbf{r}_w\}$
   Update the microscopic deformation: $\{\mathbf{w}\} = \{\mathbf{w}\} + \{\Delta\tilde{\mathbf{w}}\}$
   In the BFA, everytime solve for update χ : $\mathbf{K}_\chi[\chi^Q] = [\mathbf{r}_\chi]$
   (In the MBFA, sometime solve for update χ_0 : $\mathbf{K}_\chi[\chi^Q] = [\mathbf{r}_\chi]$)
   Compute : $\mathbf{S}, \{\mathbf{r}_u\} - \mathbf{K}_{uw}\{\Delta\tilde{\mathbf{w}}\}$
3. Communication using MPIallreduce
   Assemble : $\mathbf{S}, \{\mathbf{r}_u\} - \mathbf{K}_{uw}\{\Delta\tilde{\mathbf{w}}\}$
4. Solve
   $$\mathbf{S}\{\Delta\mathbf{u}\} = \{\mathbf{r}_u\} - \mathbf{K}_{uw}\{\Delta\tilde{\mathbf{w}}\}$$
5. Update the vatiables
   $\{\mathbf{u}\} = \{\mathbf{u}\} + \{\Delta\mathbf{u}\}$
   In the BFA, $\{\mathbf{w}\} = \{\tilde{\mathbf{w}}\} - [\chi]\Delta\overline{\mathbf{F}}$
   In the MBFA, $\{\mathbf{w}\} = \{\tilde{\mathbf{w}}\} - [\chi_0]\Delta\overline{\mathbf{F}}$
6. if $|\{\mathbf{r}_w\}| < Tol, |\{\mathbf{r}_u\}| < Tol$, then
   Next step GOTO 1
   else
   Next iteration GOTO 2

(b)

1. Set calculation condition
2. Perform in a unit cell at each quadrature point by paraell computation
   Compute : $\mathbf{K}_{ww}, \{\mathbf{r}_w\}$
   Solve : $\mathbf{K}_{ww}\{\Delta\tilde{\mathbf{w}}\} = \{\mathbf{r}_w\}$
   Update the microscopic deformation: $\{\mathbf{w}\} = \{\mathbf{w}\} + \{\Delta\tilde{\mathbf{w}}\}$
   (Sometime solve for update χ_0 : $\mathbf{K}_\chi[\chi^Q] = [\mathbf{r}_\chi]$)
   Compute : $\mathbf{K}_{uu}, \mathbf{K}_{u\alpha}, \mathbf{K}_{\alpha u}, \mathbf{K}_{\alpha\alpha}, \{\mathbf{r}_\alpha\}, \{\mathbf{r}_u\}$
3. Communication using MPIallreduce
   Assemble : $\mathbf{K}_{uu}, \mathbf{K}_{u\alpha}, \mathbf{K}_{\alpha u}, \mathbf{K}_{\alpha\alpha}, \{\mathbf{r}_\alpha\}, \{\mathbf{r}_u\}$
4. Solve
   $$\begin{bmatrix}\mathbf{K}_{\alpha\alpha} & \mathbf{K}_{\alpha u}\\ \mathbf{K}_{u\alpha} & \mathbf{K}_{uu}\end{bmatrix}\begin{Bmatrix}\Delta\alpha\\ \Delta\mathbf{u}\end{Bmatrix} = \begin{Bmatrix}\mathbf{r}_\alpha\\ \mathbf{r}_u\end{Bmatrix}$$
5. Update the vatiables
   $$\{\mathbf{u}\} = \{\mathbf{u}\} + \{\Delta\mathbf{u}\}$$
   $$\{\alpha\} = \{\alpha\} + \{\Delta\alpha\}$$
   $$\{\mathbf{w}\} = \{\tilde{\mathbf{w}}\} - [_0\chi]\{\Delta\alpha\}$$
6. if $|\{\mathbf{r}_w\}| < Tol, |\{\mathbf{r}_u\}| < Tol$, then
   Next step GOTO 1
   else
   Next iteration GOTO 2

(c)

Fig. 2 Algorithms for parallel computation using the message passing interface (MPI). **a** Algorithms for GA. **b** Algorithms for BFA and MBFA. **c** Algorithms for MSA

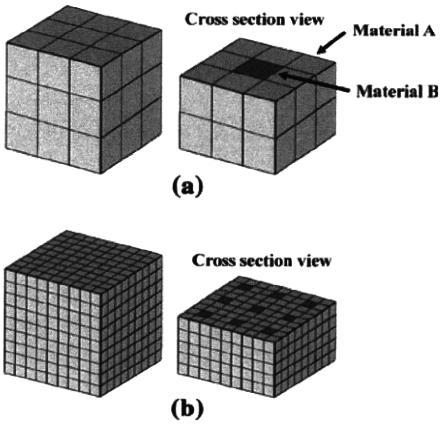

Fig. 3 FE models used in the calculation time evaluation. **a** 64 nodes, 27 elements (minimum size). **b** 1000 nodes, 729 elements

Table 1 Material constants for the models used in the calculation time evaluation

	E[kPa]	ν
Material A in Fig. 3	0.1	0.3
Material B in Fig. 3	10.0	0.3

of both scales simultaneously. Moreover, in the MBFA and MSA the straightforward update of S is avoided. We performed a 20% tensile test using the mesh of the block shown in Fig. 3 as the microstructure, and a single 8-node element as the macrostructure. The minimum size of the mesh is constructed from 27 ($3 \times 3 \times 3$) elements as shown in Fig. 3a, with the stiffness of the center element different from the rest.

The NDOF is adjusted by adding the same number of minimum units in each direction (e.g. Fig. 3b). The St. Venant hyperelastic material is used for the constitutive equations

$$W = \frac{1}{2}\lambda(tr\mathbf{E})^2 + \mu\mathbf{E} : \mathbf{E}, \tag{82}$$

$$\mathbf{S} = \frac{\partial W}{\partial \mathbf{E}} = (\lambda\mathbf{I} \otimes \mathbf{I} + 2\mu\mathbf{I}) : \mathbf{E} = \mathbf{C} : \mathbf{E}, \tag{83}$$

$$l_{ijkl} = \delta_{ik}\delta_{jl}, \tag{84}$$

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \quad \mu = G = \frac{E}{2(1+\nu)}, \tag{85}$$

where $\mathbf{I}$ is a second order identity tensor, λ and μ are Lame constants, and E and ν are, respectively, Young's modulus and Poisson's ratio. $\mathbf{C}$ is a fourth order constant elasticity tensor and the relationship between $\boldsymbol{S}$ and $\boldsymbol{E}$ is linear. The material constants are given in Table 1. If the material constants are assumed to be uniform in the microstructure, the solutions obtained by these methods agree completely with the theoretical solution in the case of infinitesimal deformation. We have thus confirmed the correctness of the formulations

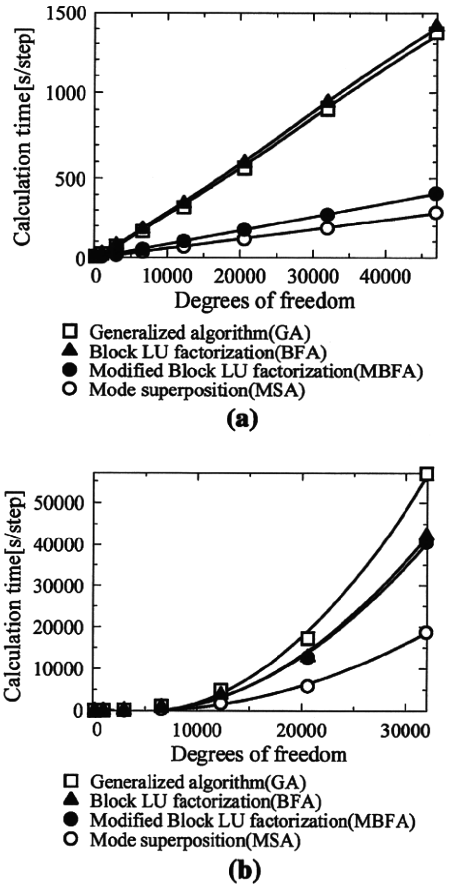

Fig. 4 Comparison of calculation time. **a** Iterative solver. **b** Direct solver

and computer programs. The relationship between calculation time and NDOF of the microstructure for each algorithm is depicted in Fig. 4 for cases where an iterative solver (ILU preconditioned GMRES method) or direct solver (skyline method) are used. The breakdown of calculation time and number of iterations in each process, where the NDOF of the microstructure is 31,944, are given in Table 2. Convergence is judged to occur when the root sum square reaches 1×10^{-10} times the initial value. As shown in Fig. 4, the calculation time of the BFA using an iterative solver slightly exceeds that of the GA. In the BFA, equilibrium of the microstructure is not required within an iteration for the sake of computational efficiency, but this may result in more iterations compared with the GA. Table 2 illustrates this, in that the BFA requires 5 iterations whereas the GA requires 4. We need to calculate the characteristic deformation to update the Schur-Complement; this means that the nine different right-hand sides of Eq. (33) must be solved in the case of the iterative solver. Consequently, the increased number of iterations results in a deterioration in the performance of the BFA. If a direct solver is used instead, the result of the LU decomposition

Table 2 Calculation time and number of iterations in each process

	Calculation of χ (s)	Calculation of equilibrium in the microscopic model (s)	Total (s)	No. of iterations
a. Iterative solver				
GA	670	211	906	4
BFA	805	107	942	5
MBFA	17	195	273	11
MSA	21	113	186	5
b. Direct solver				
GA	19,816	36,949	56,791	4
BFA	23,689	18,492	42,212	5
MBFA	451	40,050	40,557	11
MSA	457	18,237	18,746	5

of the stiffness matrix can be reused in the computation of the characteristic deformation, and as a result, the computational cost of the BFA is 20% less than that of the GA, despite the BFA using one more iteration than the GA. On the other hand, the MBFA is much faster than both the BFA and the GA when an iterative solver is used, although in the case of a direct solver there is almost no difference between the MBFA and BFA. This is because the calculation cost of the mode update in the BFA is similar to that of the increased iterations in the MBFA. However, convergence in the MBFA is slow as shown in Fig. 5 and it is anticipated that for strong nonlinear problems, convergence may not be achieved at all. Here, the characteristic deformation was approximated by that obtained at the beginning of the analysis. Convergence can be improved by incorporating more frequent updates, at the expense of CPU time. Compared with these methods, the MSA always exhibits excellent speed and convergence. The advantage of the MSA is twofold. First, the approximation of the Schur-Complement is more accurate than in the MBFA, i.e., Eq. (80) gives a superior approximation than Eq. (72). Second, in an iteration of the MSA, the microscopic equation is solved in Eq. (75) in terms of $\Delta\alpha$ as well as in Eq. (74) with Eq. (69). These advantages should contribute to an accelerated convergence of the microscopic equation as well as the macroscopic equation. Although the St. Venant hyperelastic material assumed here has relatively weak nonlinearity, the number of iterations does not increase up to a large deformation range, even with the initial characteristic deformation being used for ${}_0\chi$. However, a periodic update of ${}_0\chi$ is required in the case of strong nonlinear material, such as an elasto-plastic material. If ${}_0\chi$ is updated at every iteration, the homogenized tangent stiffness corresponds to the exact value as described in Eq. (80).

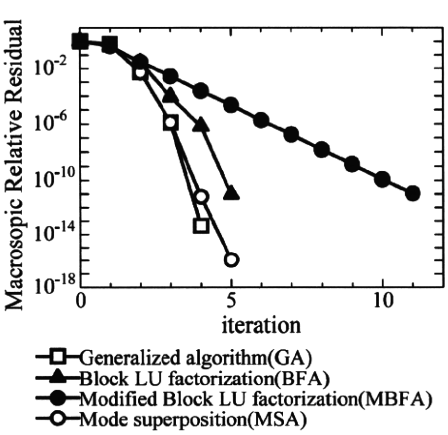

Fig. 5 Comparison of convergence property

4.2 Application to ventricle–cardiomyocyte analysis with parallel computation

As seen in Table 2, most of the calculation cost relates to the microscopic equilibrium (Eq. (69)) when the MBFA or MSA is used. In addition, as the NDOF of the microscopic model becomes larger, the computational cost for the macroscopic model becomes more negligible. It is therefore crucial to decrease the time for microscopic calculation in real problems. Since Eq. (69) holds independently at each macroscopic quadrature point, parallel computation is effective in the homogenization method. In other words, microscopic models can be distributed equally to the available cores, and this directly accelerates the microscopic calculation according to the number of cores. Moreover, since the memory is shared by fewer microscopic models in the parallel computation, a greater NDOF of the microstructure can be handled.

Figures 6 and 7 show a simplified human cardiomyocyte–ventricle model, to which we have applied the MSA. As shown on the left and in the center of Fig. 7 a simplified cardiomyocyte model is constructed with extracellular and intracellular matrices and gap junctions. The total NDOF is 20385. If the models are arrayed periodically in the three directions as depicted on the right of the figure, a fairly accurate imitation of a microgram of real tissue is obtained. The

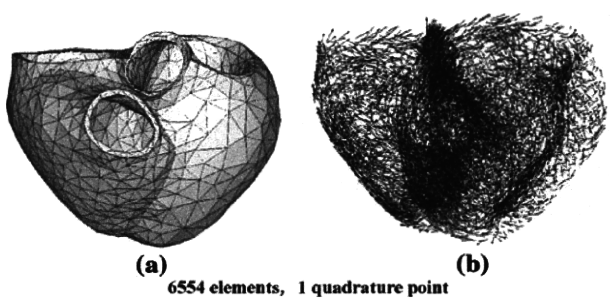

Fig. 6 FE meshes of ventricles as macroscopic model. **a** FE mesh. **b** Fiber orientation

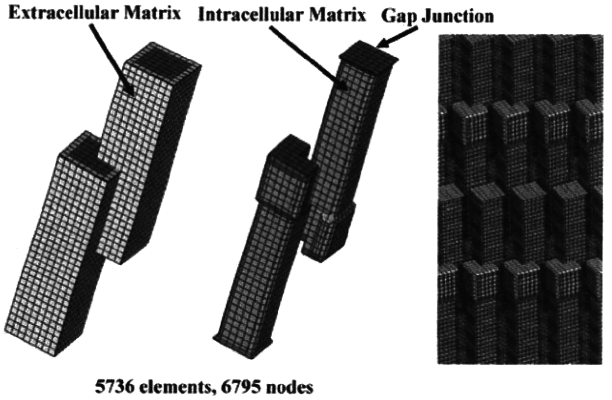

Fig. 7 FE meshes of cardiomyocytes as microscopic model

Table 3 Material constants for the cell model

	C_1	C_2	κ
Gap Junction	1×10^5	1×10^5	2×10^5
Intracellular Matrix	1×10^4	1×10^4	2×10^4
Extracellular Matrix	1×10^3	1×10^3	2×10^3

Mooney–Rivlin material using reduced invariants is adopted for the constitutive equation, with the material constants listed in Table 3.

$$W = C_1(\tilde{I}_c - 3) + C_2(\tilde{II}_c - 3) + \frac{\kappa}{2}U^2 \tag{86}$$

$$\tilde{I}_c = \frac{I_c}{III_c^{1/3}}, \quad \tilde{II}_c = \frac{II_c}{III_c^{2/3}} \tag{87}$$

$$U = J - 1, \tag{88}$$

where U is the volumetric strain energy function and κ is the bulk modulus. The ventricle model is constructed based on CT imaging, with fiber directions distributed from -90 to 60 degrees, relative to the plane perpendicular to the long axis of the ventricle. The fiber direction denotes the long axis (z-axis) of the cardiomyocyte model, and therefore, proper rotation is taken into consideration for each micromodel. In this model, the intracellular matrix is defined as a function of a parameter that represents the excitation of the myocyte, and this parameter is varied at every time step to represent the transient contraction force. The homogenization method is applied to 6,554 elements covering the greater region of the ventricle, whereas the conventional Mooney–Rivlin hyperelastic constitutive law is assumed for the limited regions at the base and apex. To reduce the computational cost, a single myocyte (micromodel) is assigned to each finite element of the ventricle, i.e., 6,554 cardiomyocyte models are embedded in the ventricle model. The total NDOF, including those of the ventricles, amounts to 133,609,263.

The computer used was an IBM Blade Center consisting of 336 Power6 (4.0 GHz) processors. Considering the size of

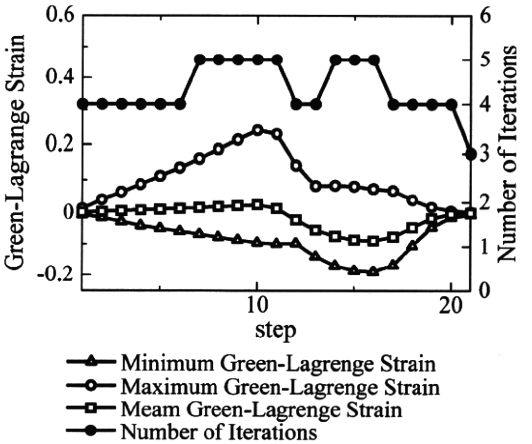

Fig. 8 Green–Lagrenge strains and nunber of iterations at each step

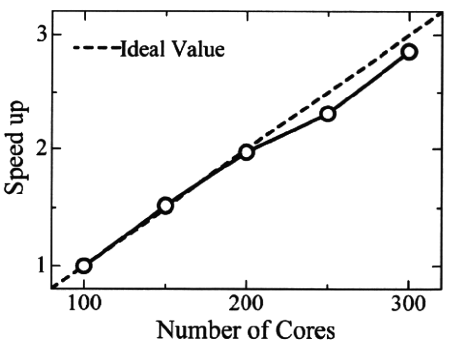

Fig. 9 Speed up in parallel computation

the problem, convergence was deemed to occur when the root sum square of the macro and micro residuals was 1×10^{-5} times less than the initial value. Figure 8 shows the maximum, the minimum and the mean Green–Lagrange strains throughout the ventricle model, and the number of iterations of the MSA at each time step. The strains are measured in the fiber direction. Although the maximum and the minimum strains reach $\pm 20\%$, an excellent convergence property was observed. Figure 9 shows the resulting scalability, by depicting the speed up rate relative to a hundred-core computation. As shown in this figure, satisfactory parallel performance was obtained by the proposed algorithm. The deformations of the ventricle and a representative myocyte in the diastole and the systole are exemplified in Fig. 10. The CPU time for the completion of a cardiac cycle was about 24 h when using 300 cores. The proposed method thus allows us to deal with large-scale problems.

5 Conclusion

To reduce the computational cost of the nonlinear homogenization method, the theoretical framework was reassessed

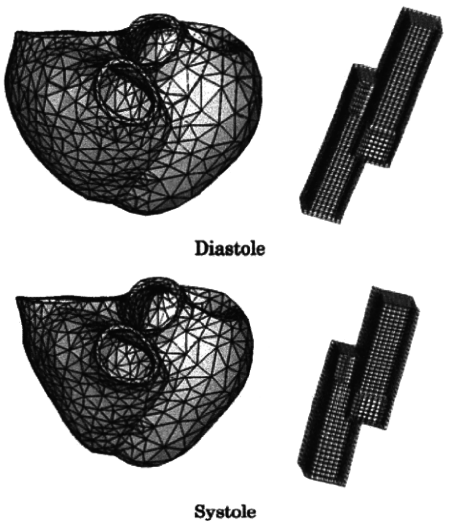

Fig. 10 Deformations of macro and microstructures in diastole and systole

from the perspective of block LU factorization of the micro–macro coupled equation. Based on the relation between the characteristic deformation and the Schur-Complement as the homogenized tangent stiffness, a couple of approximation methods were introduced, namely, a simple scheme to reuse the old characteristic deformation (MBFA) and a sophisticated method based on the mode-superposition method (MSA) developed by our group. It is noted that accuracy is preserved in these approximation methods by incorporating the equilibrium conditions in both scales. Then, using a simplified FE model, the conventional algorithm (GA), a relative algorithm originating from the block LU factorization (BFA), the MBFA, and the MSA were compared and discussed. Of these methods, the MSA was found to be the best. Then, using the MSA, a large-scale human ventricle–cardiomyocyte simulation was performed on an IBM Blade Center consisting of 336 Power6 processors, and good parallel performance was demonstrated. We plan to use the proposed homogenization algorithm in a whole-heart simulation on a massively parallel computer in the near future.

Acknowledgments Our current research on homogenization method stems from Reference [10] coauthored by Hirohisa Noguchi. We again recognize his enthusiasm and contribution in a broad range of computational mechanics. This work was supported by Core Research for Evolutional Science and Technology, Japan Science and Technology Agency.

References

1. Terada K, Hori M, Kyoya T, Kikuchi N (2000) Simulation of the multi-scale convergence in computational homogenization approaches. Int J Solids Struct 37:2285–2311
2. Hollister SJ, Kikuchi N (1992) Comparison of homogenization and standard mechanics analyses for periodic porous composites. Comput Mech 10:73–95
3. Breuls RGM, Sengers BG, Oomens CWJ, Bouten CVC, Baaijens FPT (2002) Predicting local cell deformations in engineered tissue constructs: a multilevel finite element approach. ASME J Biomech Eng 124:198–207
4. Krassowska W, Pilkington TC, Ideker RE (1990) Potential distribution in three-dimensional periodic myocardium: Part I. Solution with two-scale asymptotic analysis. IEEE Trans Biomed Eng 37:252–266
5. Terada K, Kikuchi N (2001) A class of general algorithms for multi-scale analyses of heterogeneous media. Comput Methods Appl Mech Eng 190:5427–5464
6. Matsui K, Terada K, Yuge K (2004) Two-scale finite element analysis of heterogeneous solids with periodic microstructures. Comput Struct 82:593–606
7. Takano N, Ohnishi Y, Zako M, Nishiyabu K (2001) Microstructure-based deep-drawing simulation of knitted fabric reinforced thermoplastics by homogenization theory. Int J Solids Struct 38:6333–6356
8. Rohan E (2003) Sensitivity strategies in modelling heterogeneous media undergoing finite deformation. Math Comput Simul 61:261–270
9. Moulinec H, Suquet P (1998) A numerical method for computing the overall response of nonlinear composites with complex microstructure. Comput Methods Appl Mech Eng 157:69–94
10. Yamamoto M, Hisada T, Noguchi H (2001) Homogenization method for large deformation problem by using characteristic deformation mode superposition. Trans Jpn Soc Mech Eng 67:1877–1884 (in Japanese)
11. Yamamoto M, Hisada T (2002) A homogenization method for nearly incompressible materials by using characteristic deformation mode superposition. Trans Jpn Soc Mech Eng A 45:596–602
12. Okada J, Hisada T (2006) Homogenization method for mixed finite element analysis of incompressible hyper-elastic materials. In: JSCES e06: Proceedings of the 11th computational engineering conference, Osaka, Japan, pp 79–82 (in Japanese)
13. Okada J, Hisada T (2006) Homogenization method for mixed finite element analysis of incompressible viscoelastic materials. In: Proceedings of the 19th JSME computational mechanics conference, Nagoya, Japan, pp 45–46 (in Japanese)
14. Okada J, Hisada T (2009) Study on compressibility control of hyperelastic material for homogenization method using mixed finite element analysis. J Comput Sci Technol 3(1):89–100
15. Okada J, Washio T, Hisada T (2009) Nonlinear homogenization algorithms with low computational cost. J Comput Sci Technol 3(1):101–114
16. Miehe C, Schroder J, Schotte J (1999) Computational homogenization analysis in finite plasticity. Simulation of texture development in polycrystalline materials. Comput Methods Appl Mech Eng 171:387–418
17. Allaire G (1992) Homogenization and two-scale convergence. SIAM J Math Anal 23:1482–1518
18. Terada K, Kikuchi N (2003) Introduction to the method of homogenization. Maruzen (in Japanese)
19. Yamada T, Matsui K (2007) Numerical procedure for nonlinear multiscale problems based on block Newton method. Proc Conf Comput Eng Sci 12:309–312 (in Japanese)

SIAM REVIEW
Vol. 52, No. 4, pp. 717–743

A Parallel Multilevel Technique for Solving the Bidomain Equation on a Human Heart with Purkinje Fibers and a Torso Model*

Takumi Washio†
Jun-ichi Okada†
Toshiaki Hisada‡

Abstract. In this paper, we present a multigrid method and its implementation on parallel computers to solve the bidomain equation that appears in excitation propagation analysis of the human heart with the torso. The bidomain equation is discretized with the finite element method on a composite mesh composed of a fine voxel mesh around the heart and a coarse voxel mesh covering the torso. The extracellular potential problem on the torso is formulated as a variational problem with a constraint at the interface of the fine and coarse meshes. We show that this formulation naturally satisfies the conservation property of the electric currents and fits into the multilevel adaptive solution technique framework. We also present our special treatment of the Purkinje fiber network in the multigrid algorithm where it is modeled as multiway branching lines connected to the nodes in the voxel mesh of the heart. A parallel implementation of the proposed multigrid algorithm on distributed memory computers is presented and its performance is evaluated using real-life applications.

Key words. bidomain equation, excitation propagation, torso, multigrid, composite mesh

AMS subject classifications. 65N50, 65N30, 92C30

DOI. 10.1137/100798429

1. Introduction.

The bidomain equation is a widely accepted mathematical model for describing the excitation propagation and external stimulation of heart tissue [18, 23]. It consists of two partial differential equations (PDEs) coupled to ordinary differential equations (ODEs). The two PDEs describe the intra- and extracellular electrical circuits, respectively, that are separated by cell membranes. The ODEs incorporate the behavior of the cell membranes (capacitance of the membrane and the switching of several kinds of ion channels) to determine the currents between the two circuits. A simplified representation of this bidomain equation can be seen in Figure 1.1. The upper panel shows an array of cardiac myocytes which are electrically connected to each other by gap junctions and are surrounded by the extracellular

*Published electronically November 8, 2010. This paper originally appeared in *SIAM Journal on Scientific Computing*, Volume 30, Number 6, 2008, pages 2855–2881, and was supported by Core Research for Evolutional Science and Technology, Japan Science and Technology Agency.
http://www.siam.org/journals/sirev/52-4/79842.html

†Graduate School of Frontier Sciences, University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa, Chiba, 277-0882, Japan (washio@sml.k.u-tokyo.ac.jp, okada@sml.k.u-tokyo.ac.jp).

‡Graduate School of Frontier Sciences, University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-0033, Japan (hisada@mech.t.u-tokyo.ac.jp).

liquids. Due to the capacitance effect of the cell membranes (red lines), there are discontinuities of the electrical potential field across the membranes in this microscopic view. In the bidomain equation, an electrical analogy of this situation is made as depicted in the lower left panel of Figure 1.1. Here, we assume that there are two continuous potential fields, one for the extracellular potential ϕ_e and the other for the intracellular potential ϕ_i, which are separated by the capacitor (cell membranes) and connected by the ion channels. We also assume that the intracellular and extracellular potential fields are connected by the conductances corresponding to the gap junctions and the extracellular liquids, respectively. As the conductance inside the myocyte is much larger than that of the gap junctions, it can be ignored for the intracellular potential field. For computation of the ion channel current I_{ion}, the Luo–Rudy model [13] and the Noble model [21] have been implemented for ventricular myocytes, while the Nattel model [6] was used for atrial myocytes in our simulator. For example, the majority of ion channel currents in the Luo–Rudy model are given as

$$I_{ion} = I_{Na} + I_{Si} + I_K + I_{K1} + \cdots, \tag{1.1}$$

$$I_{Na} = \bar{g}_{Na} \cdot m^3 \cdot h \cdot j \cdot (V_m - E_{Na}), \tag{1.2}$$

$$I_{Si} = \bar{g}_{Si} \cdot d \cdot f \cdot (V_m - E_{Si}), \ldots, \tag{1.3}$$

where I_{Na}, I_{Si}, $I_K, \ldots$ are the electrical currents through the individual ion channels driven by the differences of the transmembrane potential $V_m = \phi_i - \phi_e$ and the potentials in the equilibrium for the individual ions $(E_{Na}, E_{Si}, \ldots)$. The coefficients of these potential differences are given by the products of the maximal values of the conductances $(\bar{g}_{Na}, \bar{g}_{Si}, \ldots)$ and the gating variables $(m, h, j, d, f, \ldots)$. A gating variable x is time-varying and is governed by the differential equation

$$\frac{dx}{dt} = \alpha_x(V_m)(1 - x) - \beta_x(V_m)x, \tag{1.4}$$

where the functions α_x and β_x are given for each gating variable x. Note that there are other types of ion channel currents that depend on the intracellular ion concentrations. Further, there are more complex models in which some ion channels are influenced by the metabolic reactions [5] and mechanical stimulations [11]. Our heart simulator (UT heart simulator [24]) attempts to integrate these electrophysiological phenomena and couple them with the mechanical function of the heart as a blood pump. In cardiac myocytes, Ca^{2+} plays an important role in electrical activity and in controlling the contraction force of myofibrils. As depicted in the lower right panel of Figure 1.1, Ca^{2+} release and uptake at the sarcoplasmic reticulum (SR) are also incorporated in the aforementioned cardiac myocyte models.

In this paper, we utilize the bidomain equation in which the geometries and material properties of the heart and the torso are incorporated. As depicted in Figure 1.2, let Ω_H be the domain of the heart muscle and Γ_H the boundary of Ω_H. The explanations for the subdomains Ω_C and Ω_P outside Ω_H are given later in this section. The bidomain equation for the intracellular potential ϕ_i and the extracellular potential ϕ_e coupled through the transmembrane potential $V_m = \phi_i - \phi_e$ on the heart muscle is given by

$$-\nabla \cdot \boldsymbol{\sigma}_i \nabla \phi_i = -\beta I_m \text{ on } \Omega_H, \tag{1.5}$$

$$\vec{n}_H \cdot \boldsymbol{\sigma}_i \nabla \phi_i = 0 \text{ on } \Gamma_H, \tag{1.6}$$

$$-\nabla \cdot \boldsymbol{\sigma}_e \nabla \phi_e = \beta I_m \text{ on } \Omega_H, \tag{1.7}$$

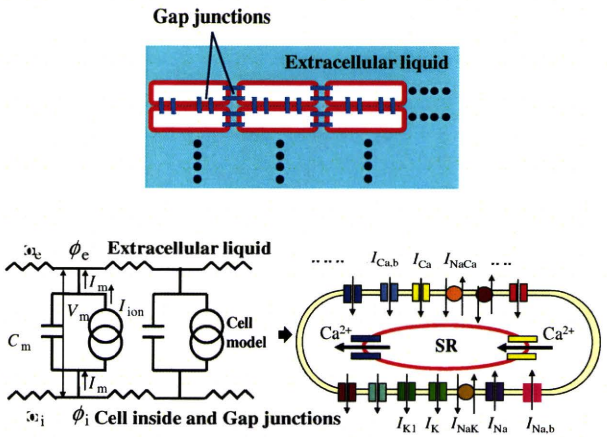

Fig. 1.1 *The array of cardiac cells (upper panel), its electrical analogy (lower left panel), and the ion channels at the cell membrane and at the SR inside the cell (lower right panel).*

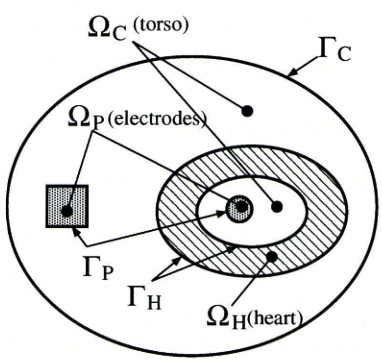

Fig. 1.2 *Two-dimensional image of the heart domain Ω_H, the torso outside the heart Ω_C, the electrodes Ω_P, and their boundaries.*

$$\vec{n}_H \cdot \boldsymbol{\sigma}_e \nabla \phi_e = J_H \text{ on } \Gamma_H, \tag{1.8}$$

$$I_m = C_m \frac{\partial V_m}{\partial t} + I_{ion}(V_m, \boldsymbol{S}) \text{ on } \Omega_H. \tag{1.9}$$

Here, $\boldsymbol{\sigma}_i$ and $\boldsymbol{\sigma}_e$ are the intra- and extracellular conductivity tensors, respectively, β is the surface-to-volume ratio of the tissue, C_m is the membrane capacitance per unit area, and I_m is the transmembrane current per unit area. The transmembrane current I_m is a combination of a capacitive current $C_m \partial V_m / \partial t$ and an ionic current $I_{ion}(V_m, \boldsymbol{S})$ passing through several kinds of ion channels, where $\boldsymbol{S}$ is a state vector composed of gating variables. J_H is a current entering the extracellular domain of the heart through Γ_H, and $\vec{n}_H$ is the outward normal vector on Γ_H.

The intra- and extracellular conductivity tensors of the myocardium (heart muscle) are anisotropic. They are given by

$$\boldsymbol{\sigma} = c_f \boldsymbol{n}_f \otimes \boldsymbol{n}_f + c_s \boldsymbol{n}_s \otimes \boldsymbol{n}_s + c_n \boldsymbol{n}_n \otimes \boldsymbol{n}_n, \tag{1.10}$$

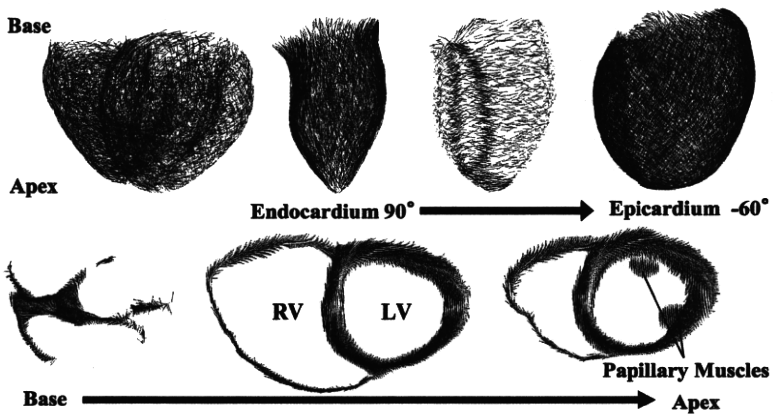

Fig. 1.3 *Fiber directions in the ventricles. The orientation of a fiber varies from $+90°$ to $-60°$ relative to the circumferential direction in a plane perpendicular to the transmural direction from the endocardiac to the epicardiac layer.*

Table 1.1 *Conductivities [mS/cm] of the myocardial cells and the surface-to-volume ratio of the tissue β [1/cm].*

Cells	Intra (f, s, n)	Extra (f, s, n)	β
Ventricle	7.2, 2.4, 2.0	6.0, 3.0, 3.0	2000
Atrium	7.0, 7.0, 7.0	7.0, 7.0, 7.0	2000

where $\boldsymbol{\sigma} = \boldsymbol{\sigma}_{\mathrm{i}}$ or $\boldsymbol{\sigma}_{\mathrm{e}}$. Here, $\boldsymbol{n}_f, \boldsymbol{n}_s$, and $\boldsymbol{n}_n$ denote the fiber, sheet, and sheet-normal directions, respectively, in the orthonormal basis [9]. The bases vary depending on the position in the heart. In general, the conductivity along the fiber direction c_f is the largest of the three. The orientations of the myocytes (fiber directions) have been modeled based on anatomical observations. Figure 1.3 depicts the distributions of the fiber directions in our simulation. The muscle fibers have a helical structure within the left ventricle (LV). Based on histological studies, fiber directions can be determined as the orientation of a fiber varies from $+90°$ to $-60°$ relative to the circumferential direction in a plane perpendicular to the transmural direction from the endocardiac to the epicardiac layer, resulting in a very realistic model for the fiber directions. Table 1.1 gives the conductivities in the three directions and the surface-to-volume ratio for our simulations. These conductivities are at least two times larger than the standard values. Due to the limitation of our computational resources, the local mesh size is set to 0.4 mm, which is still too large to obtain a proper wave propagation velocity (less than 0.1 mm is required). The conductivities have therefore been adjusted in order to obtain a usable wave propagation velocity.

Strong demands from physiological application areas require three-dimensional simulations to include the torso surrounding the heart as well. Physically, cardiac electrical activity results in a current flow in the torso via the extracellular potential field. Thus, the extracellular PDE should be extended from the heart to the torso via the boundary current J_H in (1.8).

One of the applications that requires the torso model is the simulation of an implantable cardioverter defibrillator (ICD) [12]. This device delivers a strong electrical shock to the heart through the electrodes when a fatal arrhythmia is detected, as

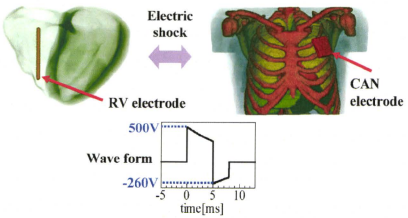

Fig. 1.4 *Two electrodes of an ICD and a typical voltage profile between them.*

depicted in Figure 1.4. By optimizing the arrangement of electrodes and the temporal profile of the electrical current, an ICD can be developed that is much more comfortable than the device currently used.

The other important application is a simulation of the electrocardiogram (ECG) routinely used in clinical practice. In order to reproduce the ECG based on the real physiological system, modeling of the so-called Purkinje fiber network is indispensable. The Purkinje fibers are located on the inner ventricular walls of the heart with large free-running sections. They are some of the specialized cardiac muscle fibers that form the impulse-conducting network of the heart, and they transmit impulses rapidly from the atrioventricular node to the ventricles. In our simulator, the DiFrancesco–Noble cell model [7] is used for the Purkinje fibers and the network is modeled from the atrioventricular node to the myocardium. With this model, we can analyze the relations between the body surface potential and the excitation propagation in the heart in a healthy condition and in various conditions involving heart disease. This research, therefore, contributes to early recognition and treatment through diagnosis with an ECG. Furthermore, a mechanism of arrhythmia that originates from the Purkinje fiber network, and the impact of the Purkinje fiber network at defibrillation, can be clarified by these simulations. Figure 1.5 shows the first half of one normal heartbeat as computed by our simulator. In the upper part, the membrane potentials on the Purkinje fiber and the heart muscle, as well as the body surface potentials, are depicted at typical times. At the bottom of the figure, the second-lead ECG obtained from this computation is depicted. This agrees well with actual clinical observations.

In our heart simulator, mechanical simulation of the beating heart, including intracavitary blood flow and movements of valve leaflets, can be also performed using the fluid-structure interaction finite element method [34, 35] (see Figure 1.6). In this simulation, the Ca^{2+} concentration data obtained from our electrophysiology model is applied to an excitation-contraction coupling model of sarcomere dynamics to compute the contraction force of every finite element of the heart muscle mesh. This results in the synchronous contraction of the heart and blood ejection. Thus far, we have been able to retrieve and present the time courses of intracavitary volumes and blood pressures and so forth for both healthy and diseased hearts. Such a simulation allows us to examine the influence of various electrophysiological phenomena on the mechanics of the heart [24].

The mathematical formulation for the extension of the extracellular potential equation to the torso is given as follows. As depicted in Figure 1.2, let Ω_C be the

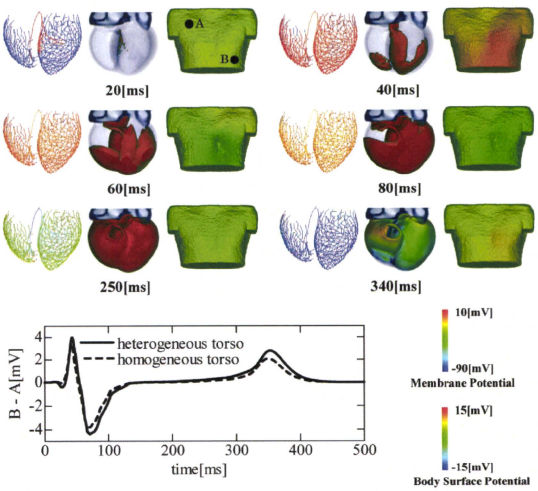

Fig. 1.5 *Membrane potentials on the Purkinje fiber network and the heart muscle, and the surface potential on the torso, at typical times. At the bottom, the second-lead ECG (difference of the potential values between points A and B) is presented for cases with the heterogeneous (rigid line) and homogeneous (broken line) torso models.*

torso domain outside the heart, Γ_C its boundaries, $\boldsymbol{\sigma}_c$ the electric conductivity tensor, and ϕ_c the potential on Ω_C. We assume that there are some regions inside Ω_C where electrodes with given potentials ϕ_p are imbedded in the case of an ICD simulation. We represent these electrode regions by Ω_P. In this situation, the following equations are further imposed on Ω_C in addition to (1.5)–(1.8):

$$-\nabla \cdot \boldsymbol{\sigma}_c \nabla \phi_c = 0 \text{ on } \Omega_C \backslash \Omega_P, \tag{1.11}$$

$$\vec{n}_H \cdot \boldsymbol{\sigma}_c \nabla \phi_c = \vec{n}_H \cdot \boldsymbol{\sigma}_e \nabla \phi_e = J_H \text{ and } \phi_c = \phi_e \text{ on } \Gamma_H, \tag{1.12}$$

$$\vec{n}_C \cdot \boldsymbol{\sigma}_c \nabla \phi_c = 0 \text{ on } \Gamma_C, \tag{1.13}$$

$$\phi_c = \phi_p \text{ on } \Omega_P. \tag{1.14}$$

As shown in Figure 1.7, the torso is composed of several kinds of organs that have different electric conductivities. Table 1.2 lists the conductivities of the organs present in our simulation. As depicted by the broken line in Figure 1.5, we observed substantially different computational results for the body surface potential when we assumed a totally homogeneous conductivity (3 mS/cm) on the torso. This indicates the importance of modeling the torso conductance accurately.

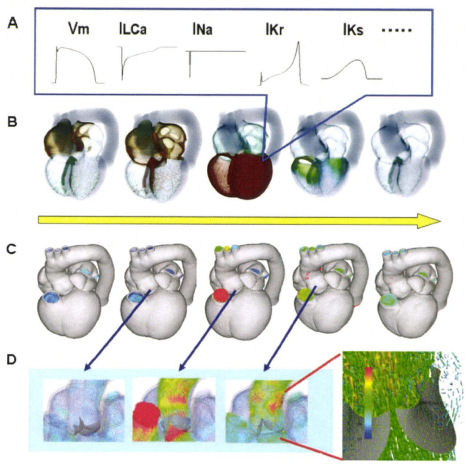

Fig. 1.6 *Computational results generated by the UT heart simulator. Time courses of the transmembrane potential and the ion channel currents at a specific point in the heart muscle* (A)*, the distribution of the transmembrane potential* (B)*, structural deformation of the heart muscle and blood flow* (C)*, and blood flow and valve leaflets at the aorta* (D)*. The yellow arrow indicates the time axis for one heartbeat.*

In this paper, we present a technique for a parallel solution that handles the bidomain problem with the torso model in a robust and efficient way. In particular, we focus on a technique to overcome the difficulty of very high spatial resolution by using a sophisticated multigrid algorithm and parallel computation techniques. The multigrid algorithm is constructed for a conservative finite element discretization of the given potential problem on a composite mesh consisting of a fine local voxel mesh around the heart and a coarse global voxel mesh covering the torso. We also briefly present strategies to resolve the difficulty of time resolution. On one hand, more sophisticated adaptive refinement techniques have been proposed in earlier studies [4, 16, 27], where the refinement is varied over time depending on the ongoing electric activity. However, on the other hand, this study is significantly more complex since the method is shown to be applicable to a problem of practical interest, whereas the earlier study merely showed the feasibility of the method on a simple two-dimensional grid. Some previous works [19, 29] also adopted a multigrid method for the bidomain problem. However, neither the composite mesh problem nor the Purkinje network is dealt with in these studies. The differences between this research and the earlier studies mentioned above are summarized below.

1. A stability analysis of a typical explicit time integration scheme of the bidomain equation [27, 29] is given.

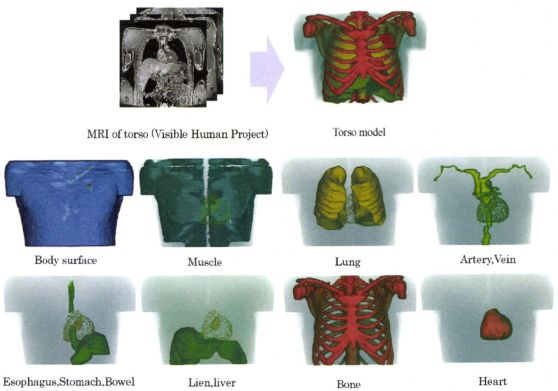

Fig. 1.7 *Organs in the torso.*

Table 1.2 *Conductivities [mS/cm] of the organs outside the heart muscle.*

Blood	7.8	Capsula cordis	15.0
Muscle	2.56	Lung	0.83
Artery, vein	6.8	Esophagus	2.0
Stomach, bowel	2.0	Lien, liver	1.67
Bone	0.06	Others	0.5

2. A conservative finite element discretization of the potential problem on the composite mesh is proposed by applying the Lagrange multiplier technique for the constraints at the interface of the local and global meshes.
3. A multigrid solution method for the discretization on the composite mesh is derived naturally by the Lagrange multiplier at the interface.
4. The conservation property of the electric currents passing through the electrodes is investigated. This is important when considering the interaction with an external electrical device, such as an ICD.
5. An efficient way of dealing with the Purkinje fiber network in the multigrid method is proposed.

In our simulator, the Purkinje fibers are modeled as one-dimensional multiway branching lines connected to the voxel nodes at these end-points, as has been reported in the literature [25]. Bidomain equations similar to (1.5), (1.6), and (1.9) are then discretized by the one-dimensional finite elements constructed on the network. In this situation, the voxel mesh can be coarsened in a standard way by exploiting its regular structure, whereas the Purkinje network is irregular. Thus, combining the Purkinje network with the multigrid algorithm is not straightforward. We also present a practical parallel implementation method on a distributed memory machine. Good performance of the implemented algorithm is demonstrated through numerical experiments

with a realistic human heart model that includes the torso. Note that unstructured grids are preferred for defibrillation studies with an ICD, since smooth boundaries are required to avoid artificial currents at the strong shock. Efficiency of the algebraic multigrid method in such cases is confirmed in the literature [17] for rabbit ventricles. In our case, we adopted the voxel mesh approach due to the simplicity of modeling and data handling, ease of parallelization, and the faster computation speed obtained by avoiding indirect memory addressing.

2. Finite Element Discretization. In this section, we introduce the temporal and spatial discretization of the bidomain equation. The weak forms of (1.5)–(1.8) on Ω_H are given by

$$\int_{\Omega_H} \nabla w_\mathrm{i} \cdot \boldsymbol{\sigma}_\mathrm{i} \nabla \phi_\mathrm{i} d\Omega = -\int_{\Omega_H} w_\mathrm{i} \beta I_\mathrm{m} d\Omega, \tag{2.1}$$

$$\int_{\Omega_H} \nabla w_\mathrm{e} \cdot \boldsymbol{\sigma}_\mathrm{e} \nabla \phi_\mathrm{e} d\Omega = \int_{\Omega_H} w_\mathrm{e} \beta I_\mathrm{m} d\Omega + \int_{\Gamma_H} w_\mathrm{e} J_H d\Gamma. \tag{2.2}$$

Furthermore, the weak form of (1.11)–(1.13) on Ω_C are given by

$$\int_{\Omega_C \backslash \Omega_P} \nabla w_\mathrm{c} \cdot \boldsymbol{\sigma}_\mathrm{c} \nabla \phi_\mathrm{c} d\Omega = -\int_{\Gamma_H} w_\mathrm{c} J_H d\Gamma, \ w_\mathrm{c} = 0 \text{ on } \Gamma_P. \tag{2.3}$$

Here, $w_\mathrm{i}, w_\mathrm{e}$, and w_c are arbitrary test functions. With the boundary conditions on Γ_H given in (1.12), we can superpose the two extracellular equations on Ω_H and Ω_C, and replace (2.2) and (2.3) by

$$\int_{\Omega} \nabla w_\mathrm{e} \cdot \boldsymbol{\sigma}_\mathrm{e} \nabla \phi_\mathrm{e} d\Omega = \int_{\Omega_H} w_\mathrm{e} \beta I_\mathrm{m} d\Omega, \tag{2.4}$$

$$\phi_\mathrm{e} = \phi_\mathrm{p}, \ w_\mathrm{e} = 0 \text{ on } \Gamma_P. \tag{2.5}$$

Here, $\Omega = \Omega_H \cup \Omega_C \backslash \Omega_P$ and Γ_P is the boundary of Ω_P. The extracellular potentials and the conductivity tensors are combined on the whole domain Ω as

$$\phi_\mathrm{e} = \begin{cases} \phi_\mathrm{e} & \text{on } \Omega_H, \\ \phi_\mathrm{c} & \text{on } \Omega_C, \end{cases} \qquad \boldsymbol{\sigma}_\mathrm{e} = \begin{cases} \boldsymbol{\sigma}_\mathrm{e} & \text{on } \Omega_H, \\ \boldsymbol{\sigma}_\mathrm{c} & \text{on } \Omega_C. \end{cases} \tag{2.6}$$

The matrix representation for a finite element discretization of the intracellular equation on Ω_H in (2.1) and the extracellular equation on Ω in (2.4) are given by

$$\boldsymbol{K}_\mathrm{i} \boldsymbol{\phi}_\mathrm{i} = -\beta \boldsymbol{I}_\mathrm{m}, \tag{2.7}$$

$$\boldsymbol{K}_\mathrm{e} \boldsymbol{\phi}_\mathrm{e} = \boldsymbol{R}_H{}^T \beta \boldsymbol{I}_\mathrm{m} - \boldsymbol{K}_\mathrm{p} \boldsymbol{\phi}_\mathrm{p}. \tag{2.8}$$

Here, we assume that the nodes on the electrodes are deleted in the vector representation $\boldsymbol{\phi}_\mathrm{e}$, and that $\boldsymbol{R}_H$ represents a restriction operator from the whole domain Ω to the subdomain Ω_H on the heart muscle. Since $\boldsymbol{R}_H$ is a simple injection on Ω_H in our case, we will omit $\boldsymbol{R}_H$ and its transpose $\boldsymbol{R}_H^T$ in the equations hereafter. For example, the matrices $\boldsymbol{R}_H^T \boldsymbol{K}_\mathrm{i}$ and $\boldsymbol{R}_H^T \boldsymbol{K}_\mathrm{i} \boldsymbol{R}_H$ will simply be represented by $\boldsymbol{K}_\mathrm{i}$.

2.1. Explicit Time Integration Scheme. In order to establish a stable time integration scheme, (2.7) and (2.8) are rewritten as

$$\boldsymbol{K}_\mathrm{i} \boldsymbol{V}_\mathrm{m} + \beta \boldsymbol{I}_\mathrm{m} + \boldsymbol{K}_\mathrm{i} \boldsymbol{\phi}_\mathrm{e} = \boldsymbol{0}, \tag{2.9}$$

$$\boldsymbol{K}_\mathrm{i} \boldsymbol{V}_\mathrm{m} + (\boldsymbol{K}_\mathrm{i} + \boldsymbol{K}_\mathrm{e}) \boldsymbol{\phi}_\mathrm{e} = -\boldsymbol{K}_\mathrm{p} \boldsymbol{\phi}_\mathrm{p}. \tag{2.10}$$

Equation (2.10) is obtained by adding (2.7) and (2.8) and the relation between the potentials, $\boldsymbol{V}_{\mathrm{m}} = \boldsymbol{\phi}_{\mathrm{i}} - \boldsymbol{\phi}_{\mathrm{e}}$ on Ω_H. Note that the intracellular potential disappears in this representation, and the second equation (2.10) does not contain the transmembrane current I_{m}. The system of (2.9) and (2.10) can be integrated stably along the temporal axis in either an explicit or a semi-implicit manner. In our case, we adopt the following explicit method to reduce computational costs [27]. The matrix representation of the explicit scheme is given by

$$\begin{aligned}\begin{bmatrix} \frac{\beta C_{\mathrm{m}}}{\Delta t}\boldsymbol{M}_H & \boldsymbol{0} \\ \boldsymbol{K}_{\mathrm{i}} & \boldsymbol{K}_{\mathrm{i}}+\boldsymbol{K}_{\mathrm{e}} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_{\mathrm{m}}^{t+\Delta t} \\ \boldsymbol{\phi}_{\mathrm{e}}^{t+\Delta t} \end{bmatrix} &= \begin{bmatrix} \frac{\beta C_{\mathrm{m}}}{\Delta t}\boldsymbol{M}_H - \boldsymbol{K}_{\mathrm{i}} & -\boldsymbol{K}_{\mathrm{i}} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_{\mathrm{m}}^{t} \\ \boldsymbol{\phi}_{\mathrm{e}}^{t} \end{bmatrix} \\ &\quad - \begin{bmatrix} \beta \boldsymbol{M}_H & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{K}_{\mathrm{p}} \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_{\mathrm{ion}}(\boldsymbol{V}_{\mathrm{m}}^{t}, \boldsymbol{S}^{t}) \\ \boldsymbol{\phi}_{\mathrm{p}}^{t+\Delta t} \end{bmatrix},\end{aligned} \tag{2.11}$$

where $\boldsymbol{M}_H$ is the lumped matrix on Ω_H and $\boldsymbol{\phi}_{\mathrm{p}}^{t+\Delta t}$ is a prescribed potential on the electrodes. In the semi-implicit scheme as adopted in the literature [19], the entire PDE system (except for the ODEs to compute $\boldsymbol{S}$) is solved simultaneously. This scheme requires almost quadruple the computational costs for the matrix-vector multiplication and the relaxation phase. On the other hand, the time step size Δt is usually determined in accordance with the ODE system to compute the state vector $\boldsymbol{S}$, as described later. Thus, we have adopted the explicit scheme in our simulator. If we ignore the time dependence of I_{ion} in (2.11), the stability of the explicit scheme depends on the spectral radius of the matrix:

$$\mathcal{M} = \begin{bmatrix} \boldsymbol{D}_{\mathrm{m}} & \boldsymbol{0} \\ \boldsymbol{K}_{\mathrm{i}} & \boldsymbol{K}_{\mathrm{i}}+\boldsymbol{K}_{\mathrm{e}} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{D}_{\mathrm{m}} - \boldsymbol{K}_{\mathrm{i}} & -\boldsymbol{K}_{\mathrm{i}} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \tag{2.12}$$

with

$$\boldsymbol{D}_{\mathrm{m}} = \frac{\beta C_{\mathrm{m}}}{\Delta t}\boldsymbol{M}_H. \tag{2.13}$$

The spectral radius of $\mathcal{M}$ is bounded above as described by the following theorem.

THEOREM 2.1. *Assume the spectral radius of* $\boldsymbol{D}_{\mathrm{m}}^{-1}\boldsymbol{K}_{\mathrm{i}}$ *is less than one,*

$$\rho(\boldsymbol{D}_{\mathrm{m}}^{-1}\boldsymbol{K}_{\mathrm{i}}) < 1. \tag{2.14}$$

Then the spectral radius of $\mathcal{M}$ *in* (2.12) *is bounded as*

$$\rho(\mathcal{M}) \leq 1 - \lambda_{\min}(\boldsymbol{D}_{\mathrm{m}}^{-1}\boldsymbol{K}_{\mathrm{i}})(1 - \lambda_{\max}((\boldsymbol{K}_{\mathrm{i}}+\boldsymbol{K}_{\mathrm{e}})^{-1}\boldsymbol{K}_{\mathrm{i}})). \tag{2.15}$$

Here, $\lambda_{\min}$ *and* $\lambda_{\max}$ *represent, respectively, the minimum and maximum eigenvalues of the matrices in brackets.*

A proof of the theorem is given in the literature [30]. From the theorem, we see that the scheme is stable if condition (2.14) is satisfied and the time dependence of I_{ion} is ignored. In the actual simulation, however, a fairly small time step, far smaller than the above limitation, is required to compute the ion currents $I_{\mathrm{ion}}(V_{\mathrm{m}}, \boldsymbol{S})$ in (1.9) (see [26]). Thus, we can adopt an "inner-outer" time integration strategy where the intracellular (2.9) is integrated with a small time step in the inner iterations while fixing the extracellular potential ϕ_{e}, and the extracellular potential ϕ_{e} is updated only in the outer iteration with a large time step. This strategy has been adopted extensively by many studies [27]. The algorithm is presented in Figure 2.1. Here, the superscripts denote the time indices for the variables, where T is the time index for

The inner-outer time integration scheme.
Given $\boldsymbol{V}_{\mathrm{m}}^{t_0}$
Solve $(\boldsymbol{K}_{\mathrm{i}} + \boldsymbol{K}_{\mathrm{e}})\boldsymbol{\phi}_{\mathrm{e}}^{t_0} = -\boldsymbol{K}_{\mathrm{p}}\boldsymbol{\phi}_{\mathrm{p}}^{t_0} - \boldsymbol{K}_{\mathrm{i}}\boldsymbol{V}_{\mathrm{m}}^{t_0}$
for $k = 1, \ldots, k_{\mathrm{end}}$
  $T = t_0 + (k-1)\Delta T$
  for $i = 1, \ldots, \Delta T/\Delta t$
    $t = (i-1)\Delta t + T$
    $\boldsymbol{V}_{\mathrm{m}}^{t+\Delta t} = \boldsymbol{V}_{\mathrm{m}}^{t} - \frac{\Delta t}{\beta C_{\mathrm{m}}}\boldsymbol{M}_H^{-1}\boldsymbol{K}_{\mathrm{i}}\boldsymbol{V}_{\mathrm{m}}^{t} - \boldsymbol{K}_{\mathrm{i}}\boldsymbol{\phi}_{\mathrm{e}}^{T} - \frac{\Delta t}{\beta C_{\mathrm{m}}}\boldsymbol{I}_{\mathrm{ion}}(\boldsymbol{V}_{\mathrm{m}}^{t}, \boldsymbol{S}^{t})$
  next i
  Solve $(\boldsymbol{K}_{\mathrm{i}} + \boldsymbol{K}_{\mathrm{e}})\boldsymbol{\phi}_{\mathrm{e}}^{T+\Delta T} = -\boldsymbol{K}_{\mathrm{p}}\boldsymbol{\phi}_{\mathrm{p}}^{T+\Delta T} - \boldsymbol{K}_{\mathrm{i}}\boldsymbol{V}_{\mathrm{m}}^{T+\Delta T}$
next k

Fig. 2.1 *The inner-outer time integration scheme.*

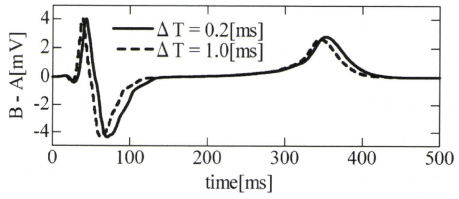

Fig. 2.2 *The second-lead ECG for two different time steps.*

solving the extracellular potential problem and t is the time index for integrating the membrane potential. At each inner iteration, the ODE to compute the state vector $\boldsymbol{S}$ is solved by the explicit Euler scheme. The time step δt (typically $10{\sim}100\,\mu$s) is flexibly varied in time and space depending on the ongoing electric activity as described in the literature [27]. At each outer iteration, a solution to the potential problem on Ω is required. The time steps ΔT and Δt are adjusted flexibly depending on the simulation conditions and the desired accuracy of the solution. For example, in Figure 2.2, the two curves represent the second-lead ECG results as explained in section 1 for time steps $\Delta T = 0.2$ ms and $\Delta T = 1$ ms, respectively. Though both curves show a similar tendency and magnitude of fluctuation, substantial phase lag is observed.

Even when adopting a large time step for ΔT in the above inner-outer time integration scheme, the speedup of the solution to the extracellular potential problem is the key issue for the overall performance. Henceforth, we will focus on an efficient discretization of and solution method for the potential problem.

2.2. Spatial Discretization of a Composite Voxel Mesh. Hereafter, $\boldsymbol{\sigma}$ denotes $\boldsymbol{\sigma}_{\mathrm{e}} + \boldsymbol{\sigma}_{\mathrm{i}}$ and ϕ denotes ϕ_e for notational convenience. The weak form corresponding to (2.10) is given by

$$\int_{\Omega} \nabla w \cdot \boldsymbol{\sigma} \nabla \phi d\Omega = -\int_{\Omega_H} \nabla w \cdot \boldsymbol{\sigma}_{\mathrm{i}} \nabla V_{\mathrm{m}} d\Omega,$$
$$w = 0 \text{ and } \phi = \phi_{\mathrm{p}} \text{ on } \Gamma_P. \tag{2.16}$$

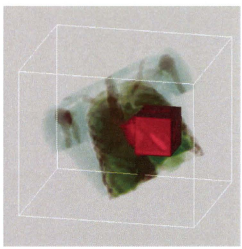

Fig. 2.3 *The fine local voxel mesh domain around the heart and the global coarse voxel mesh domain covering the torso.*

To discretize the above equation, we adopt a uniform three-dimensional voxel mesh because this allows the input data to be prepared easily, and it facilitates the implementation of the multigrid solver and the parallelization of the code. For the intracellular equations (1.5) and (1.9) on the heart, a fine spatial resolution ($0.15\,\mathrm{mm} \sim 0.4\,\mathrm{mm}$) is required to attain sufficient accuracy. On the other hand, such a fine resolution is unnecessary for the extracellular equation (2.4) outside the heart. The above fact naturally leads to the use of two meshes with different spatial resolutions. One is a finer mesh on the local rectangular parallelepiped domain around the heart. The other is a coarser mesh on the global rectangular parallelepiped domain covering the whole torso (see Figure 2.3). Henceforth, we refer to the former mesh as the local mesh and to the latter as the global mesh. We assume that the nodes in the global mesh on the local rectangular parallelepiped domain are given at the same positions as those in the local mesh, and the interval ratio of the global to local mesh is a power of two. Note that it is not necessarily exactly two, as in the usual local refinement. For example, four or eight can also be handled in our framework.

In order to discretize (2.16) on the composite mesh, we apply the Lagrange multiplier method for the constraints at the interface of the local and global meshes. Therefore, we start with a variational formulation of the problem. The energy functional for the formulation is given by

$$\mathcal{E}(\phi) = \int_{\Omega} \frac{1}{2} \nabla\phi \cdot \boldsymbol{\sigma} \nabla\phi d\Omega + \int_{\Omega_H} \nabla\phi \cdot \boldsymbol{\sigma}_{\mathrm{i}} \nabla V_{\mathrm{m}} d\Omega,$$
$$\phi = \phi_{\mathrm{p}} \text{ on } \Gamma_P. \tag{2.17}$$

The extremum of the energy functional in (2.17) satisfies (2.16). Let Ω^L and Ω^G be the local and global meshes, respectively. In particular, Ω^L is a local refinement of Ω^G. In the following, Ω^L and Ω^G are identified either as the domains covered by the meshes or as the sets of nodes in them. Let E^L and E^G be the sets of finite elements contained in Ω^L and Ω^G, respectively. Let Ω^G_L and $\overline{\Omega^G_L}$ be the subsets of Ω^G on Ω^L and outside of Ω^L, respectively. Let E^G_L and $\overline{E^G_L}$ be the subsets of E^G which lie in Ω^G_L and $\overline{\Omega^G_L}$, respectively (see Figure 2.4). We represent an interpolated function ϕ^L