Appendix L

Periodic Boundary Conditions[1]

L.1 General

The discussion here adopts the viewpoints described [1, 2, 3]. The potentially complex micro-scale behavior at a material point (M in the figure) is integrated into the macroscale structural model using the linear/nonlinear response of representative volume elements (RVEs), see Figs 1 and 2.

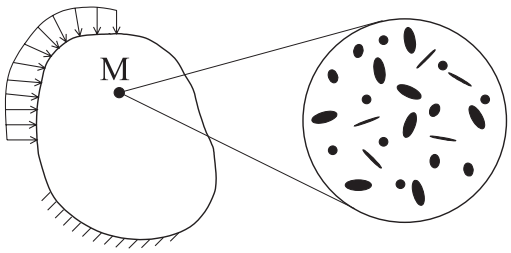


Fig. 1. Concept of a complex microstructure at macroscale point M. Figure taken from [1].

When the same micro-scale material structure appears repeatedly in the vicinity of M, RVE concepts with periodic boundary conditions may become applicable. Figure 2 illustrates a two-dimensional configuration. Extension to three-dimensions is considerably more complex and is the focus of this Appendix. Further, the undeformed boundaries of the repeated edges (2D) and surfaces (3D) are not required to be straight/planar. Researchers in this field, including [1] and references therein, discuss the challenges in defining the correct RVE for non-trivial material/structural configurations.

The remainder of this Appendix develops equations for PBCs and the procedures to impose them on general 3D RVEs using the features available in WARP3D. This appendix focuses exclusively on displacement-based periodic boundary conditions for three-dimensional RVEs and does not address traction or mixed boundary conditions. Section L.2 develops the mathematical form of periodic boundary conditions for 3D RVEs. Section L.3 explains the practical implementation in WARP3D using dummy nodes, rigid link elements and multi-point constraints. The detailed descriptions here for 3D configurations are somewhat subtle and rarely found in the published literature. Section L.4 provides an example RVE model to illustrate the form of 3D, periodic boundary conditions.

[1] See the directory RVE_support in the WARP3D distribution for example problems and a Python program to generate the many thousands of multi-point constraints required to enforce periodic boundary conditions for 3D representative volume element (RVEs) models.

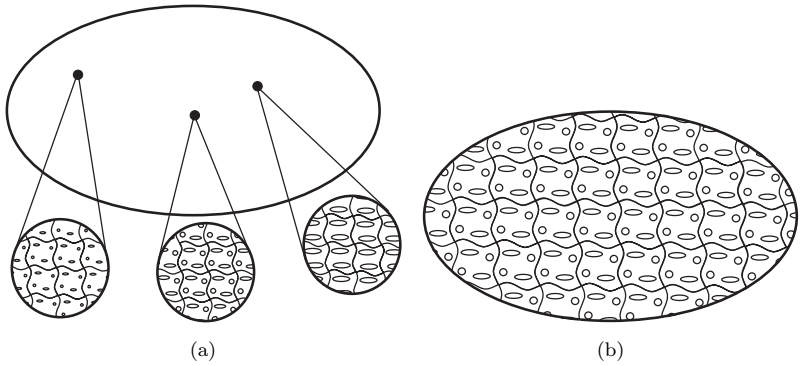


Fig. 2. Local micro-structure with repeated periodicity in the macroscale model. Figure taken from [1].

L.2 Development of PBC Equations

The periodic boundary conditions in a Cartesian framework have the form

$$u_i\left(x_1, x_2, x_3\right) = \bar{\varepsilon}_{ik}\, x_k + u_i^*\left(x_1, x_2, x_3\right), \quad i, k = 1, 2, 3 \tag{L.1}$$

where $\bar{\varepsilon}_{ik}$ denotes the known, average strain tensor at material point M to be imposed on the RVE. Apparently, P.M. Suquet [4] was among the first to state the PBCs in the above form and has been adopted subsequently by others. With $\bar{\varepsilon}_{ij} = \frac{1}{2}\left(\partial u_i/\partial x_j + \partial u_j/\partial x_i\right)$, the development here for multi-point constraint (MPC) equations to enforce the PBCs does not require a symmetric strain tensor in the above equation. Adoption of the strain tensor to describe loading is actually for algorithmic convenience rather than a fundamental requirement.

For a simple RVE with a homogeneous structure loaded only, for example, by $\bar{\varepsilon}_{12} = a$ and $\bar{\varepsilon}_{21} = b$, WARP3D computes/outputs $\gamma_{xy} = a + b$, even when $a \neq b$. The values a, b cannot equal zero but one can be made very small to approximate zero if needed.

$u_i^*\left(x_1, x_2, x_3\right)$ denotes the periodic displacement field identical across adjacent RVEs. u_i^* is most often unknown and dependent on the RVE loading. In WARP3D, constraints are incremental and applied in each load step. See Chapter 2 for more discussion. Equation (L.1) does not uniquely define the displacement field unless one reference point is fixed. Additional constraints may be required to prevent rigid rotations.

To proceed, the boundary surfaces of the RVE must be defined to have parallel pairs such that displacements of the $(\pm)$ pairs are

$$u_i^{j+} = \bar{\varepsilon}_{ik}\, x_k^{j+} + u_i^* \tag{L.2a}$$

$$u_i^{j-} = \bar{\varepsilon}_{ik}\, x_k^{j-} + u_i^*\ . \tag{L.2b}$$

where $j+$ and $j-$ denote the j^{th} pair of parallel, opposing surfaces of the RVE. Because u_i^* is periodic, the values of u_i^* at matching points on j^+ and j^- are equal. Combining the above, we can write

$$u_i^{j+} - u_i^{j-} = \bar{\varepsilon}_{ik}\left(x_k^{j+} - x_k^{j-}\right) = \bar{\varepsilon}_{ik}\Delta x_k^j\ . \tag{L.3}$$

For RVEs with parallelepiped shapes, Δx_k^j are constants. Then,

$$\mathbf{u}^{j+} - \mathbf{u}^{j-} = \bar{\boldsymbol{\varepsilon}}\, \Delta \mathbf{x}^j \ . \tag{L.4}$$

In expanded form (no sum on j),

$$u_1^{j+} - u_1^{j-} = \bar{\varepsilon}_{11}\, \Delta x_1^j + \bar{\varepsilon}_{12}\, \Delta x_2^j + \bar{\varepsilon}_{13}\, \Delta x_3^j, \tag{L.5a}$$

$$u_2^{j+} - u_2^{j-} = \bar{\varepsilon}_{21}\, \Delta x_1^j + \bar{\varepsilon}_{22}\, \Delta x_2^j + \bar{\varepsilon}_{23}\, \Delta x_3^j, \tag{L.5b}$$

$$u_3^{j+} - u_3^{j-} = \bar{\varepsilon}_{31}\, \Delta x_1^j + \bar{\varepsilon}_{32}\, \Delta x_2^j + \bar{\varepsilon}_{33}\, \Delta x_3^j \ . \tag{L.5c}$$

In the above, the $\pm$ opposing parallel surfaces are not required to be planar, but they must have matching points, *i.e.*, finite element *node pairs*[2]. Figure 2 shows an example of this configuration.

Common cases adopt an RVE having the shape of a rectangular prism (see Fig. 3). On the $j = 1$ surfaces (normal to x_1), for example, $\Delta x_2^1 = \Delta x_3^1 = 0$. Let $L_x = \Delta x_1^1$ for convenience. Similarly, $L_y = \Delta x_2^2,\ L_z = \Delta x_3^3$.

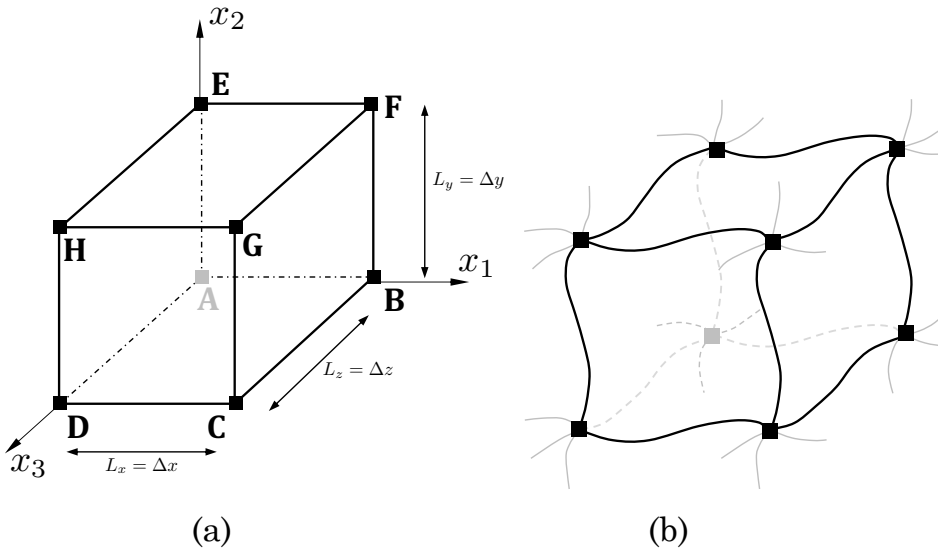


Fig. 3. (a) Labelling of vertexes, edges and faces for rectangular prism RVE. (b) Example periodic displacements [5].

The rectangular prism in Fig. 3 has sets of nodes: 8 vertices, 12 edges and 6 faces. Each node on the boundaries of the finite element model for the RVE belongs to only one of these sets. The periodic boundary conditions (PBCs) are developed separately for nodes belonging to each set. Face nodes present the simplest case. Those nodes are incident on a single pair of ($\pm$) of matching surfaces – edge nodes are incident on 2 initially orthogonal surfaces, vertex nodes on 3 orthogonal surfaces. This makes the definition of 3D PBCs considerably more complex than for 2D models.

[2]Pairs of nodes located on opposite faces of the RVE whose in-plane coordinates are identical. They represent the same physical point in the periodic material and are coupled by the periodic displacement constraints.

L.2.1 Face Nodes

The description here, Fig. 3, adopts the face notation in [3], that is less complex than the notation in [2]; both references reach identical equations to enforce the PBCs. The $\pm x_1$ faces are labelled (FGCB,EHDA); $\pm x_2$ faces are labelled (FEHG,BADC); and $\pm x_3$ faces are labelled (GHDC,FEAB). The notation FGCB represents the sequence of nodes on the x_1^+ face counterclockwise order. Using the general form in Eq. (L.5) for the rectangular prism shape and L_x, L_y, L_z notation, the PBCs for the face nodes then have the simple form because only one component of Δx_j is non zero:

$$u_i^{FGCB} - u_i^{EHDA} - L_x\,\bar{\varepsilon}_{i1} = 0, \tag{L.6a}$$

$$u_i^{FEHG} - u_i^{BADC} - L_y\,\bar{\varepsilon}_{i2} = 0, \quad i = 1,2,3 \tag{L.6b}$$

$$u_i^{GHDC} - u_i^{FEAB} - L_z\,\bar{\varepsilon}_{i3} = 0\ . \tag{L.6c}$$

L.2.2 Edge Nodes

Consider nodes along edge F-G (not including the vertex nodes). These nodes are incident on two of the (+) faces: FGCB ($L_x = \Delta x_1$) and FEHG. ($L_y = \Delta x_2$). The corresponding (−) faces are EHDA and BADC. Thus the (−) *edge pair* for F-G is A-D.[3] Applying Eq. (L.5) for this edge pair (where $i = 1,2,3$),[4]

$$\text{Edges FG} \leftrightarrow \text{AD:} \quad u_i^{FG} - u_i^{AD} - L_x\,\bar{\varepsilon}_{i1} - L_y\,\bar{\varepsilon}_{i2} = 0\ . \tag{L.7}$$

Nodes on edge H-G are incident on the two (+) faces: GHDC ($L_z = \Delta x_3$) and FEHG. ($L_y = \Delta x_2$). The corresponding (−) faces are FEAB and BADC. Thus the (−) *edge pair* for H-G is A-B. Applying Eq. (L.5) for this edge pair (where $i = 1,2,3$),

$$\text{Edges HG} \leftrightarrow \text{AB :} \quad u_i^{HG} - u_i^{AB} - L_y\,\bar{\varepsilon}_{i2} - L_z\,\bar{\varepsilon}_{i3} = 0\ . \tag{L.8}$$

And for edge pair G-C and E-A,

$$\text{Edges GC} \leftrightarrow \text{EA :} \quad u_i^{GC} - u_i^{EA} - L_x\,\bar{\varepsilon}_{i1} - L_z\,\bar{\varepsilon}_{i3} = 0\ . \tag{L.9}$$

Now consider edge B-C incident on (x_1^+) face FGCB and (x_2^-) face BADC. The corresponding edge pair is E-H incident on (x_1^-) face EHDA and (x_2^+) face FEHG. Applying Eq. (L.5) for this edge pair,

$$\text{Edges BC} \leftrightarrow \text{EH :} \quad u_i^{BC} - u_i^{EH} - L_x\,\bar{\varepsilon}_{i1} + L_y\,\bar{\varepsilon}_{i2} = 0\ . \tag{L.10}$$

Following this pattern for the remaining two edge pairs,

$$\text{Edges FE} \leftrightarrow \text{CD :} \quad u_i^{FE} - u_i^{CD} - L_y\,\bar{\varepsilon}_{i2} + L_z\,\bar{\varepsilon}_{i3} = 0\ . \tag{L.11}$$

$$\text{Edges FB} \leftrightarrow \text{HD :} \quad u_i^{FB} - u_i^{HD} - L_x\,\bar{\varepsilon}_{i1} + L_z\,\bar{\varepsilon}_{i3} = 0\ . \tag{L.12}$$

[3] The sign pattern in each equation arises directly from the signed coordinate differences $\Delta x = x_1^{(+)} - x_1^{(-)}$, $\Delta y = x_2^{(+)} - x_2^{(-)}$, $\Delta z = x_3^{(+)} - x_3^{(-)}$. These differences may be positive or negative depending on vertex ordering.

[4] The identification of edge vertices, *e.g.*, F-G or G-F, is immaterial since no direction /orientation is implied by the ordering. These differences may be positive or negative depending on vertex ordering.

L.2.3 Vertex Nodes

The pairs for vertex nodes are G-A, F-D, H-B, C-E. Each vertex is incident on three orthogonal faces. For example, node G is incident on (x_1^+, x_2^+, x_3^+) faces. Its pair node A is incident on (x_1^-, x_2^-, x_3^-) faces. Node F is incident on (x_1^+, x_2^+, x_3^-) faces; node D is incident on (x_1^-, x_2^-, x_3^+) faces, and so on for H-B, C-E. The terms on the right side of Eq. (L.5) are

$$\mathrm{G} \rightarrow \mathrm{A}: \quad \Delta x = L_x, \quad \Delta y = L_y, \quad \Delta z = L_z, \tag{L.13a}$$
$$\mathrm{F} \rightarrow \mathrm{D}: \quad \Delta x = L_x, \quad \Delta y = L_y, \quad \Delta z = -L_z, \tag{L.13b}$$
$$\mathrm{H} \rightarrow \mathrm{B}: \quad \Delta x = -L_x, \quad \Delta y = L_y, \quad \Delta z = L_z, \tag{L.13c}$$
$$\mathrm{C} \rightarrow \mathrm{E}: \quad \Delta x = L_x, \quad \Delta y = -L_y, \quad \Delta z = L_z. \tag{L.13d}$$

Then using Eq. (L.5),

$$u_i^G - u_i^A - L_x\,\bar{\varepsilon}_{i1} - L_y\,\bar{\varepsilon}_{i2} - L_z\,\bar{\varepsilon}_{i3} = 0, \tag{L.14a}$$
$$u_i^F - u_i^D - L_x\,\bar{\varepsilon}_{i1} - L_y\,\bar{\varepsilon}_{i2} + L_z\,\bar{\varepsilon}_{i3} = 0, \tag{L.14b}$$
$$u_i^H - u_i^B + L_x\,\bar{\varepsilon}_{i1} - L_y\,\bar{\varepsilon}_{i2} - L_z\,\bar{\varepsilon}_{i3} = 0, \tag{L.14c}$$
$$u_i^C - u_i^E - L_x\,\bar{\varepsilon}_{i1} + L_y\,\bar{\varepsilon}_{i2} - L_z\,\bar{\varepsilon}_{i3} = 0\ . \tag{L.14d}$$

L.3 Dummy-Node and Link-Element Method

Equations (L.6)–(L.14) specify displacement differences between matching pairs of nodes on the RVE boundary. Directly enforcing these non-homogeneous constraints in a finite element code complicates formation of the global equation system.

WARP3D instead uses the method of dummy-node pairs and rigid-link elements (Section 3.15). The non-homogeneous terms are shifted to displacements applied at auxiliary (dummy) nodes, so that all MPCs become homogeneous. This greatly simplifies assembly of the equations and is compatible with the WARP3D existing constraint infrastructure. A similar procedure is often used to enforce PBCs in Abaqus. Figure 4 shows the setup for a simplified application of the dummy node-rigid link element method. A multi-point constraint equation in WARP3D format could have the form,

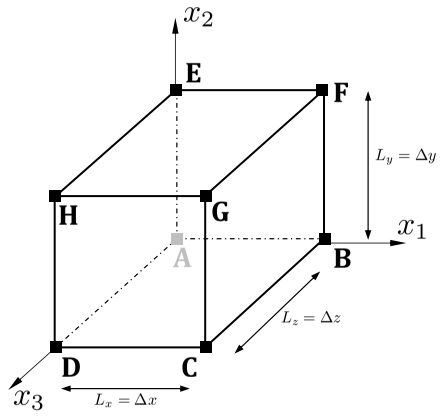
$$\mathsf{27\ 1.0\ w\ -\ 20\ 1.0\ w\ -49\ 1.0\ w = 0} \tag{L.15}$$

The link2 element is connected to "dummy" nodes 49 and 50. For convenience these nodes are often defined well outside the RVE – their actual (global) position is immaterial. Spring stiffness values in the element in global (x, y, z) directions are set to a sufficiently large value. Absolute constraints are imposed on node 50, *e.g.*,

$$\mathsf{50\ \ u\ = 0\ \ v = 0\ \ w = 0.0025} \tag{L.16}$$

Face Nodes	**Vertex Nodes**	**Edge Nodes**
$u_i^{\mathrm{FGCB}} - u_i^{\mathrm{EHDA}} - L_x\,\bar{\varepsilon}_{i1} = 0$	$u_i^{\mathrm{G}} - u_i^{\mathrm{A}} - L_x\,\bar{\varepsilon}_{i1} - L_y\,\bar{\varepsilon}_{i2} - L_z\,\bar{\varepsilon}_{i3} = 0$	FG $\leftrightarrow$ AD: $u_i^{\mathrm{FG}} - u_i^{\mathrm{AD}} - L_x\,\bar{\varepsilon}_{i1} - L_y\,\bar{\varepsilon}_{i2} = 0$
$u_i^{\mathrm{FEHG}} - u_i^{\mathrm{BADC}} - L_y\,\bar{\varepsilon}_{i2} = 0$	$u_i^{\mathrm{F}} - u_i^{\mathrm{D}} - L_x\,\bar{\varepsilon}_{i1} - L_y\,\bar{\varepsilon}_{i2} + L_z\,\bar{\varepsilon}_{i3} = 0$	HG $\leftrightarrow$ AB: $u_i^{\mathrm{HG}} - u_i^{\mathrm{AB}} - L_y\,\bar{\varepsilon}_{i2} - L_z\,\bar{\varepsilon}_{i3} = 0$
$u_i^{\mathrm{GHDC}} - u_i^{\mathrm{FEAB}} - L_z\,\bar{\varepsilon}_{i3} = 0$	$u_i^{\mathrm{H}} - u_i^{\mathrm{B}} + L_x\,\bar{\varepsilon}_{i1} - L_y\,\bar{\varepsilon}_{i2} - L_z\,\bar{\varepsilon}_{i3} = 0$	GC $\leftrightarrow$ EA: $u_i^{\mathrm{GC}} - u_i^{\mathrm{EA}} - L_x\,\bar{\varepsilon}_{i1} - L_z\,\bar{\varepsilon}_{i3} = 0$
	$u_i^{\mathrm{C}} - u_i^{\mathrm{E}} - L_x\,\bar{\varepsilon}_{i1} + L_y\,\bar{\varepsilon}_{i2} - L_z\,\bar{\varepsilon}_{i3} = 0$	BC $\leftrightarrow$ EH: $u_i^{\mathrm{BC}} - u_i^{\mathrm{EH}} - L_x\,\bar{\varepsilon}_{i1} + L_y\,\bar{\varepsilon}_{i2} = 0$
		FE $\leftrightarrow$ CD: $u_i^{\mathrm{FE}} - u_i^{\mathrm{CD}} - L_y\,\bar{\varepsilon}_{i2} + L_z\,\bar{\varepsilon}_{i3} = 0$
		FB $\leftrightarrow$ HD: $u_i^{\mathrm{FB}} - u_i^{\mathrm{HD}} - L_x\,\bar{\varepsilon}_{i1} + L_z\,\bar{\varepsilon}_{i3} = 0$

Table 1. Node equations for faces, vertices, and edges in periodic boundary conditions.



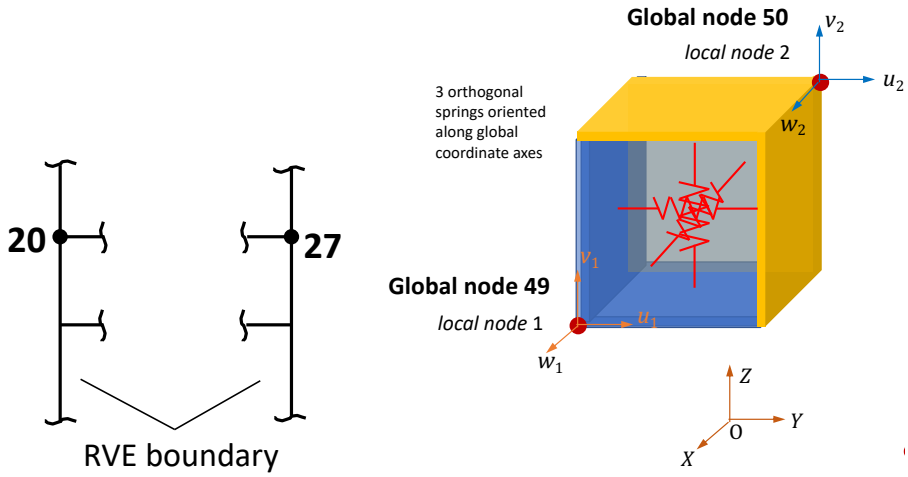


Fig. 4. Example demonstrating use of dummy-nodes and rigid-link element to enable definition of homogeneous multi-point constraints.

Given the large spring stiffnesses for the link element, node 49 will have the same displacements as node 50. Node 49 in Eq. (L.15) thus has the same w displacement as node 50, while maintaining the required homogeneous form. The stiffness should be several orders of magnitude larger than the effective RVE stiffness while remaining numerically stable.

L.4 Example Problem

See Fig. 5 for the setup. Figure 6 lists the node coordinates and element connectivities. The analysis is linear-elastic for simplicity. The model has 8, l3disop elements (8-node isoparametrics); 4 link2 elements; 27 nodes for the prism; 4 dummy-node pairs (8 total nodes); and a linear-elastic material.

The prism dimensions are set to $L_x = 1, L_y = 2, L_z = 4$ to emphasize their appearance in the multi-point constraint equations. The macroscale strain tensor is

$$\bar{\boldsymbol{\varepsilon}} = \begin{bmatrix} \bar{\varepsilon}_{11} & \bar{\varepsilon}_{12} & \bar{\varepsilon}_{13} \\ \bar{\varepsilon}_{21} & \bar{\varepsilon}_{22} & \bar{\varepsilon}_{23} \\ \bar{\varepsilon}_{31} & \bar{\varepsilon}_{32} & \bar{\varepsilon}_{33} \end{bmatrix} = \begin{bmatrix} 0.1 & 0.2 & 0.5 \\ 0.2 & 0.0 & 0.3 \\ 0.5 & 0.3 & 0.0 \end{bmatrix} \tag{L.17}$$

where values are chosen for convenience to make obvious the results are correct.

One dummy-node pair and link2 element for each non-zero strain component are defined Here the macroscale shear strains are symmetric; only one dummy-node pair and link element needed. The (2,2) and (3,3) strain values could be non-zero with addition of 2 more link2 elements and 2 more dummy-node pairs.

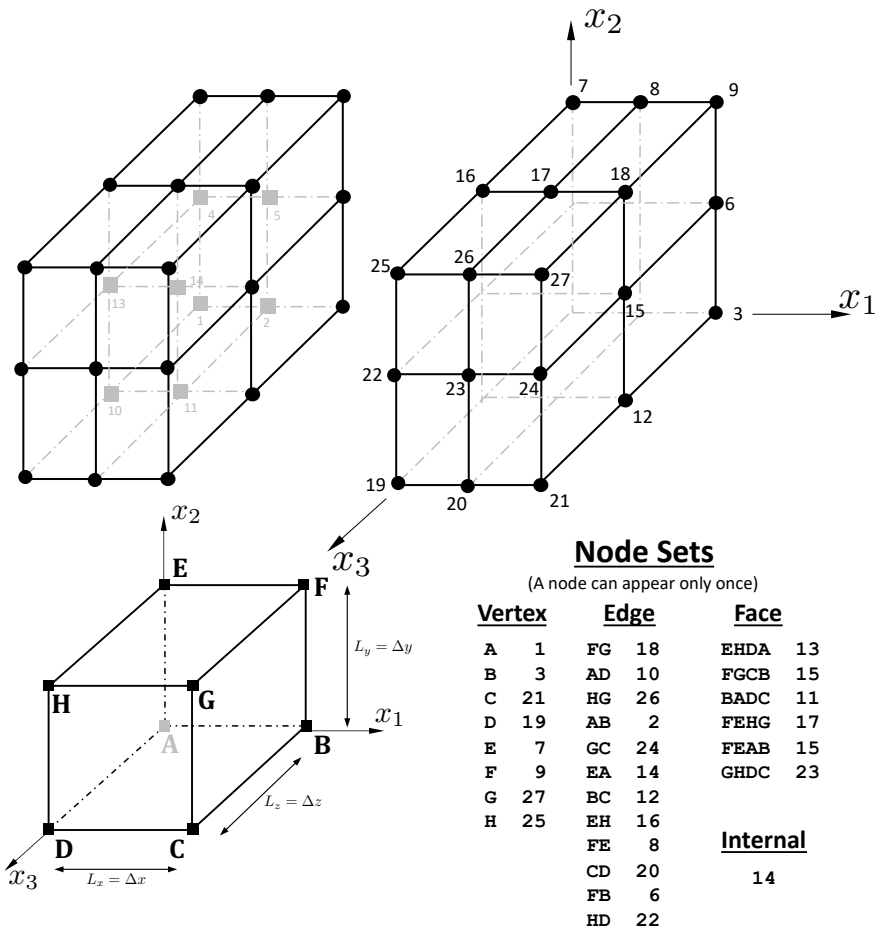


Fig. 5. Small example problem.

Coordinates for dummy nodes 28-35 are set at 10.0, 10.0, 10.0. The values are immaterial and do not affect the solution. link2 stiffnesses are set at 10^{10} – preliminary analyses showed this value is sufficient to make displacements at both nodes of link2 elements identical.

The full WARP3D input file is shown in Fig. 7. Figure 8 shows the included file of constraints data.

L.4.1 User generation of constraints

For this simple problem, it is feasible to generate the absolute and MPC constraints manually.

Recommended steps:

- Decide upon absolute constraints to prevent 3 rigid-body translations and 3 rotations (rigid-body constraints). For this loading in Eq. (11), only node 1 needs to be constrained ($u = v = w = 0$). The non-zero imposed displacements (here from macroscale strains, $\bar{\varepsilon}_{ij}$) are sufficient to prevent the three rigid-rotations.

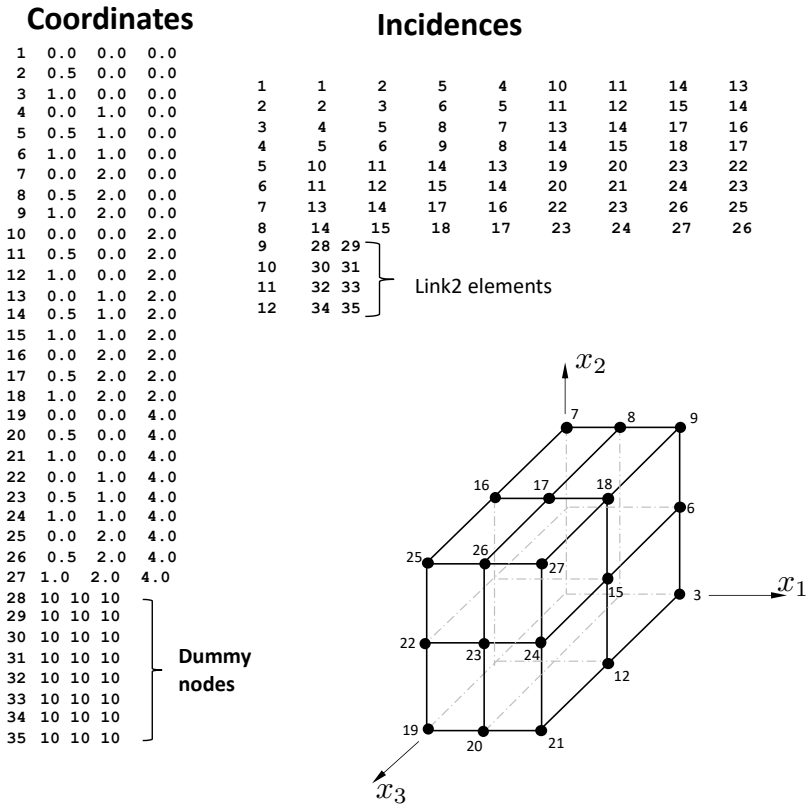


Fig. 6. Coordinates and incidences for the WARP3D input file. Put into file named coords_incid.inp.

- Start with all equations in Table 1. Eliminate all the zero terms from the zero imposed $\bar{\varepsilon}_{ij}$ values and constraints to remove rigid-body motions.
- Some of the MPCs to enforce the periodic boundary conditions may then have only 1 term remaining. The MPC has become an absolute constraint and must be moved to the set of absolute constraints. In WARP3D input, absolute constraints may not be intermixed with MPCs.

The imposed displacements for dummy-node loading become:

$$\begin{aligned}
&\text{Node 29}: \quad u = 0.1,\ v = 0.0,\ w = 0.0 \ \leftarrow \bar{\epsilon}_{11} \\
&\text{Node 31}: \quad u = 0.2,\ v = 0.2,\ w = 0.0 \ \leftarrow \bar{\epsilon}_{12} = \bar{\epsilon}_{21} \\
&\text{Node 33}: \quad u = 0.5,\ v = 0.0,\ w = 0.5 \ \leftarrow \bar{\epsilon}_{13} = \bar{\epsilon}_{31} \\
&\text{Node 35}: \quad u = 0.0,\ v = 0.3,\ w = 0.3 \ \leftarrow \bar{\epsilon}_{23} = \bar{\epsilon}_{32}
\end{aligned} \tag{L.18}$$

```
material steel
    properties mises e 30000  nu 0.3 n_power 10 yld_pt 60.e10  $
linear-elastic
!
 material rve_link
     properties link stiff_link 1.0e10 mass_link 0.0
!
 structure prism
!
 number of nodes 35 elements 12
!
 elements
    1-8 type l3disop linear material steel order 2x2x2 bbar,
                          center_output short
    9 10 11 12 type link2 material rve_link
!
*echo off
 *input from file "coords_incid.inp"
*echo on
!
blocking automatic
!
  *input from file ”rve_constraints.inp"
!
 loading test
  nonlinear
    step 1 constraints 1.0
!
 nonlinear analysis parameters $ only those really needed here
   solution technique sparse direct
   time step 1.0e06
   maximum iterations 5 $ global Newton iterations
   minimum iterations 1
   convergence test norm res tol 0.001
   batch messages off
   trace solution on
!
 output model flat patran convention text file "model"
!
 compute displacements for loading test step  1
 output wide strains 1-8
 output wide stresses 1-12
 output wide displacements 1-35
 output flat text displacements     $ for ParaView visualization
 output flat text element stresses  $ for ParaView visualization
!
stop
```

Fig. 7. WARP3D input file for the example problem.

L.4.2 Program to generate the constraints

For RVE models with many thousands of nodes, manual generation of the multi-point constraints becomes impractical.

A Python script has been developed to perform this task for RVEs in the shape of rectangular prisms. The directory RVE_support contains the script, a small user manual, and files for an example problem. With a small input file to describe vertex nodes, dimensions and loading of the prism ($\bar{\varepsilon}_{ij}$), the script generates a ready-to-use file of all constraints input for inclusion in the WARP3D input file.

L.5 Nonlinear Analyses

The constraints defined to WARP3D are applied incrementally in each load(time) step. Constraints consist of two types. Absolute, *e.g.*, 42 v = 0.2, and homogeneous, multipoint equations, *e.g.*, 16 1.0 u − 22 1.0 u − 28 1.0 u − 32 4.0 u = 0.

```
constraints
 1 u 0.0 v 0.0 w 0.0
 29 u 0.1
 31 u 0.2 v 0.2
 33 u 0.5 w 0.5
 35 v 0.3 w 0.3
!
multipoint
 15 1.0 u  - 13 1.0 u  - 28 1.0 u = 0.
 15 1.0 v  - 13 1.0 v  - 30 1.0 v = 0.
 15 1.0 w  - 13 1.0 w  - 32 1.0 w = 0.
 17 1.0 u  - 11 1.0 u  - 30 2.0 u = 0.
 17 1.0 v  - 11 1.0 v = 0.
 17 1.0 w  - 11 1.0 w  - 34 2.0 w = 0.
 23 1.0 u  - 5 1.0 u  - 32 4.0 u = 0.
 23 1.0 v  - 5 1.0 v  - 34 4.0 v = 0.
 23 1.0 w  - 5 1.0 w = 0.
 18 1.0 u  - 10 1.0 u  - 28 1.0 u  - 30 2.0 u = 0.
 18 1.0 v  - 10 1.0 v  - 30 1.0 v = 0.
 18 1.0 w  - 10 1.0 w  - 32 1.0 w  - 34 2.0 w = 0.
 12 1.0 u  - 16 1.0 u  - 28 1.0 u  + 30 2.0 u = 0.
 12 1.0 v  - 16 1.0 v  - 30 1.0 v = 0.
 12 1.0 w  - 16 1.0 w  - 32 1.0 w  + 34 2.0 w = 0.
 24 1.0 u  - 4 1.0 u  - 28 1.0 u  - 32 4.0 u = 0.
 24 1.0 v  - 4 1.0 v  - 30 1.0 v  - 34 4.0 v = 0.
 24 1.0 w  - 4 1.0 w  - 32 1.0 w = 0.
 6 1.0 u  - 22 1.0 u  - 28 1.0 u  + 32 4.0 u = 0.
 6 1.0 v  - 22 1.0 v  - 30 1.0 v  + 34 4.0 v = 0.
 6 1.0 w  - 22 1.0 w  - 32 1.0 w = 0.
 26 1.0 u  - 2 1.0 u  - 30 2.0 u  - 32 4.0 u = 0.
 26 1.0 v  - 2 1.0 v  - 34 4.0 v = 0.
 26 1.0 w  - 2 1.0 w  - 34 2.0 w = 0.
 8 1.0 u  - 20 1.0 u  - 30 2.0 u  + 32 4.0 u = 0.
 8 1.0 v  - 20 1.0 v  + 34 4.0 v = 0.
 8 1.0 w  - 20 1.0 w  - 34 2.0 w = 0.
 27 1.0 u  - 28 1.0 u  - 30 2.0 u  - 32 4.0 u = 0.
 27 1.0 v  - 30 1.0 v  - 34 4.0 v = 0.
 27 1.0 w  - 32 1.0 w  - 34 2.0 w = 0.
 9 1.0 u  - 19 1.0 u  - 28 1.0 u  - 30 2.0 u  + 32 4.0 u = 0.
 9 1.0 v  - 19 1.0 v  - 30 1.0 v  + 34 4.0 v = 0.
 9 1.0 w  - 19 1.0 w  - 32 1.0 w  - 34 2.0 w = 0.
```

Fig. 8. Manually prepared constraints input file. The prism dimensions $L_x = 1, L_y = 2, L_z = 4$ appear in the MPCs. Highly recommended to put displacements for dummy nodes last in each equation.

A nonlinear loading condition is defined to specify the incremental loads for each step. Consider the case when loading is only by non-zero constraints. A example input sequence is:

```
...
constraints
  1 u = 0  v = 0  w = 0
  59 w = 0.25
  31 u = 0.2 v = 0.2 w = 0
  ...
  multipoint
      15 1.0 u  - 13 1.0 u  - 28 1.0 u = 0.
      15 1.0 v  - 13 1.0 v  - 30 1.0 v = 0.
      15 1.0 w  - 13 1.0 w  - 32 1.0 w = 0.
      17 1.0 u  - 11 1.0 u  - 30 2.0 u = 0.
      9 1.0 u  - 19 1.0 u  - 28 1.0 u  - 30 2.0 u  + 32 4.0 u = 0.
      9 1.0 v  - 19 1.0 v  - 30 1.0 v  + 34 4.0 v = 0.
```

```
    9 1.0 w  - 19 1.0 w  - 32 1.0 w  - 34 2.0 w = 0.
    25 1.0 u  - 3 1.0 u  + 28 1.0 u  - 30 2.0 u  - 32 4.0 u = 0.
...
loading test
    nonlinear
        step 1 constraints 1.0
        step 2 constraints 3.2
        ...
```

In the above, the multiplier 1.0 is applied to the defined absolute constraints and imposed in the solution for step 1. The multiplier has no effect on the homogeneous, multipoint equations.

In step 2, the 3.2 multiplier is applied to the absolute constraints such that the imposed, incremental displacements become 59 w = 0.8; 31 u = 0.64 v = 0.64 w = 0, and so on. Again, the multiplier 3.2 has no effect on the homogeneous, multipoint equations.

At completion of the solution for load step 2, the w displacement at node 59 is 1.05; and at node 31, u = 0.84, v = 0.84, w = 0. Further, the homogenous, multipoint equations are satisfied for the total displacements.

L.6 References

[1] V.G. Kouznetsova, M.G.D. Geers, W.A.M. Brekelmans. Computational Homogenisation For Non-Linear Heterogeneous Solids. Multiscale Modeling In Solid Mechanics - Computational Approaches Imperial College Press www.worldscibooks.com/engineering/p604.html. ISBN: 978-1-84816-307-2. 2009.

[2] MM. Shahzamanian, T. Tadepalli, AM. Rajendran, WD. Hodo, R. Mohan, R. Valisetty, PW Chung, and JJ. Ramsey. Representative Volume Element Based Modeling of Cementitious Materials. Transactions of the ASME, *Journal of Engineering Materials and Technology*, Vol. 136, 011007-1:011007-16, 2014. doi.org/10.1115/1.4025916.

[3] F. Taheri-Behrooz and E. Pourahmadi. A 3D RVE Model with Periodic Boundary Conditions to Estimate Mechanical Properties of Composites. *Structural Engineering and Mechanics*, Vol. 72, 713-722, 2019. doi.org/10.12989/sem.2019.72.6.713.

[4] P.M. Suquet. Elements of homogenization for inelastic solid mechanics. In Homogenization Techniques for Composite Media. Edited by Sanchez-Palencia and A. Zaoui). Lecture Notes in Physics, 272, 1987.

[5] F. Otero, S. Oller, X. Martinez, O.Salomón. Numerical homogenization for composite materials analysis. Comparison with other micro mechanical formulations. *Composite Structures*, Vol. 122, pp. 405-416, 2014. doi.org/10.1016/j.compstruct.2014.11.041