

文章编号:1006-2467(2019)09-1100-07

DOI: 10.16183/j.cnki.jsjtu.2019.09.012

# 基于多重节点的结构动力学显式异步长并行计算方法

马志强, 楼云锋, 李俊杰, 金先龙

(上海交通大学 机械与动力工程学院; 机械系统与振动国家重点实验室, 上海 200240)

**摘要:** 局部精细化网格会导致结构有限元动力学分析的计算时间大幅增加. 为了提高计算效率, 结合域分解法与子循环方法, 提出一种基于多重节点的结构动力学显式异步长并行计算方法. 该方法采用节点分割将模型划分为若干子分区, 子分区采用显式 Newmark 时间积分格式并根据分区单元特性选用时间步长; 相邻分区通过多重节点构成耦合区域, 小步长分区子循环过程中不涉及边界数据的插值过程. “天河二号”超算平台上的算列表明: 采用显式异步长并行方法计算结构动力学问题, 在提高计算效率的同时可以保持较高的计算精度.

**关键词:** 显式积分; 有限元分析; 多重节点; 异步并行; 精细网格

**中图分类号:** TP 391; O 242

**文献标志码:** A

## Explicit Asynchronous Time Steps Parallel Computational Method for Structural Dynamics Based on Multiple Overlapping Nodes

MA Zhiqiang, LOU Yunfeng, LI Junjie, JIN Xianlong

(School of Mechanical Engineering; State Key Laboratory of Mechanical System and Vibration, Shanghai Jiao Tong University, Shanghai 200240, China)

**Abstract:** The local refined mesh leads to a large increase of calculation time in the structural dynamic finite element analysis. In order to improve the calculation efficiency, an explicit asynchronous parallel computational method based on multiple overlapping nodes is proposed. The procedure combines domain decomposition with subcycling method. The whole structure is divided into several subdomains by node partition method and the integral step size is selected according to the internal mesh sizes of the subdomain. An explicit Newmark time integral scheme is adopted in the subdomain. Multiple overlapping nodes between different subdomains constitute the coupling region without the interpolation process in the subcycling process. Simulation results of two cases implemented on Tianhe 2 supercomputer platform show that the proposed method can improve the computational efficiency with a high computational accuracy.

**Key words:** explicit integration; finite element analysis; multiple overlapping nodes; asynchronous parallel; refined mesh

大型复杂结构的有限元分析常涉及局部冲击载荷、接触等问题, 需要针对特定分析部位划分精细化

的网格<sup>[1]</sup>. 这类问题常采用显式算法求解, 但受限于求解精度以及局部单元稳定性的要求, 整个模型需

收稿日期: 2017-11-12

基金项目: 国家重点研发计划(2016YFB0201800), 国家自然科学基金(51475287, 11772192)资助项目

作者简介: 马志强(1990-), 男, 安徽省蚌埠市人, 博士生, 研究方向为大规模结构动力学分析与并行计算.

通信作者: 金先龙, 男, 教授, 博士生导师, E-mail: jxlong@sjtu.edu.cn.

要采用较小的时间步长,这会导致计算时间大大加长<sup>[2]</sup>.因此国内外众多学者提出改进方案,期望可根据结构区域的特点,将结构分成不同属性的区域,针对单一区域选择合适的积分算法和时间步长进行动力学分析.

改进方法主要包括域分解法以及子循环方法两大类.域分解法是将计算区域分解为若干小的子区域,各个子区域单独计算.根据分区边界是否重叠可以将域分解法分为重叠与非重叠两种形式.重叠形式主要包括 Schwarz 迭代法和 Arlequin 方法<sup>[3]</sup>,非重叠形式包括 Schur 补方法和 dual Schur 方法,其区别在于处理区域边界约束的方式.张友良等<sup>[4]</sup>采用 Schur 补方法开发了岩土大规模并行计算软件. Farhat 等<sup>[5]</sup>提出了以 FETI (Finite Element Tearing and Interconnecting) 为代表的 dual Schur 方法.刘铖等<sup>[6]</sup>基于 FETI 方法研究大变形柔性多体系统动力学方程高效求解算法. Kozubek 等<sup>[7]</sup>将 FETI 方法推广到大规模非线性动力学并行仿真.

子循环算法是在一次计算步内包含两种时步即一次主步计算和多次子步计算的积分算法,主要用于解决分区步长不匹配的问题.早期研究者主要有 Smolinski<sup>[8]</sup>与 Belyschko<sup>[9]</sup>,研究对象多针对两分区的结构动力学问题. Prakash 等<sup>[10]</sup>研究了非线性结构的异步长计算,通过拉格朗日乘子约束分区边界速度连续.刘占生等<sup>[11]</sup>将子循环与预测校正方法相结合求解壁板动力响应.高双武等<sup>[12]</sup>采用子循环的紧耦合算法,分析固体火箭发动机点火瞬态工作过程.

在上述研究中,域分解法可在一定程度上提高加速比,缩短计算时间,但不同计算区域采用相同的时间步长,在处理存在局部细分网格的动力学问题时,会造成计算资源的浪费.在经典的显式子循环方法中,分区多采用单元分割,不同步长边界节点信息传递过程常涉及插值或截断<sup>[13]</sup>,这在一定程度上造成计算不稳定、降低计算精度,也会限制最大步长比的选取<sup>[14]</sup>.

为此,本文提出一种基于多重节点的显式异步长并行计算方法.该方法将多重节点引入异步长并行计算,显式小步长分区子循环过程中预测波形得以在显式重叠区域内完整传递,没有插值、截断误差,计算精度较高,不同网格区域可以采用较大的步长比.显式算法具有解耦特性(求解下一时刻的相关信息只需用到相邻节点上一时步的相关信息),使并行计算过程不需要联立求解界面方程.因此,该方法具有一般显式并行算法易于编程实现的特点,同时

相对于经典的子循环算法精度较高.

## 1 预测校正形式的显式 Newmark 求解格式

含阻尼的线弹性体结构动力学方程可以写为

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f} \quad (1)$$

式中:  $\mathbf{M}$ 、 $\mathbf{C}$  与  $\mathbf{K}$  分别代表结构质量、阻尼与刚度矩阵;  $\mathbf{f}$  代表节点外力;  $\ddot{\mathbf{u}}$ 、 $\dot{\mathbf{u}}$  和  $\mathbf{u}$  分别为节点加速度、速度与位移.

显式 Newmark 求解格式分为预测步与校正步.预测步中,第  $n+1$  时间步节点位移以及速度的预测值可以由第  $n$  时间步节点的位移、速度以及加速度表示<sup>[15]</sup>:

$$\left. \begin{aligned} \tilde{\mathbf{u}}^{n+1} &= \mathbf{u}^n + \Delta t \dot{\mathbf{u}}^n + \frac{\Delta t^2}{2}(1-2\beta)\ddot{\mathbf{u}}^n \\ \tilde{\dot{\mathbf{u}}}^{n+1} &= \dot{\mathbf{u}}^n + (1-\gamma)\Delta t \ddot{\mathbf{u}}^n \end{aligned} \right\} \quad (2)$$

式中:  $\tilde{\mathbf{u}}^{n+1}$  与  $\tilde{\dot{\mathbf{u}}}^{n+1}$  代表第  $n+1$  时间步节点位移与速度预测值;  $\beta$  与  $\gamma$  为 Newmark 积分参数;  $\mathbf{u}^n$ 、 $\dot{\mathbf{u}}^n$  与  $\ddot{\mathbf{u}}^n$  为第  $n$  时间步节点位移、速度与加速度;  $\Delta t$  为时间步长.将第  $n+1$  时间步的速度以及位移预测值带入式(1),可以得到:

$$\mathbf{M}\ddot{\mathbf{u}}^{n+1} + \mathbf{C}\tilde{\dot{\mathbf{u}}}^{n+1} + \mathbf{K}\tilde{\mathbf{u}}^{n+1} = \mathbf{f}^{n+1} \quad (3)$$

式中:  $\ddot{\mathbf{u}}^{n+1}$  为第  $n+1$  时间步节点加速度;  $\mathbf{f}^{n+1}$  为第  $n+1$  时间步节点外力.质量矩阵采用集中质量矩阵形式,此时  $\mathbf{M}$  为对角矩阵,因此不用进行刚度矩阵的分解以及求逆,由公式可以直接求解出第  $n+1$  时间步的节点加速度:

$$\ddot{\mathbf{u}}^{n+1} = \mathbf{M}^{-1}(\mathbf{f}^{n+1} - \mathbf{K}\tilde{\mathbf{u}}^{n+1} - \mathbf{C}\tilde{\dot{\mathbf{u}}}^{n+1}) \quad (4)$$

将求得的节点加速度值带入显式 Newmark 离散格式的校正步

$$\left. \begin{aligned} \mathbf{u}^{n+1} &= \tilde{\mathbf{u}}^{n+1} + \beta\Delta t^2\ddot{\mathbf{u}}^{n+1} \\ \dot{\mathbf{u}}^{n+1} &= \tilde{\dot{\mathbf{u}}}^{n+1} + \gamma\Delta t\ddot{\mathbf{u}}^{n+1} \end{aligned} \right\} \quad (5)$$

由此计算出第  $n+1$  时间步节点速度  $\dot{\mathbf{u}}^{n+1}$  以及位移校正值  $\mathbf{u}^{n+1}$ .

## 2 多重节点的显式异步长并行求解方法

### 2.1 异步长计算变量定义

结构有限元网格采用节点分割方法,将节点划分给不同分区,分割单元被相邻分区共享,并行计算过程需要共享分割单元信息.定义分区步长比:

$$m_i = \frac{\Delta t_i}{\Delta t_r} \quad (6)$$

式中:  $\Delta t_i$  为第  $i$  个分区选用的时间步长;  $\Delta t_r$  为所有

分区最小时间步长. 异步长计算过程中涉及主时间步与子循环时间步, 定义时间变量上标:

$$\mathbf{X}^{n+\frac{j}{m_i}} = \mathbf{X}(t = n\Delta t_i + j\Delta t_r, j \leq m_i) \quad (7)$$

式中:  $\mathbf{X}$  为某一变量;  $n$  为主时间步数;  $j$  为子循环时间步数. 采用节点分割策略, 定义节点布尔分割向量  $\mathbf{O}_i$ , 如果某节点在给定分区  $i$ , 则该节点在分割向量中为 1, 否则为 0. 含多重节点的分区  $i$  布尔分割向量定义为:

$$\mathbf{O}_i = \begin{bmatrix} 0 & 0 & \underbrace{1 \quad 1 \quad \cdots \quad 1 \quad 1 \quad 1}_{\text{分区 } i} & 0 & 0 \end{bmatrix} \quad (8)$$

则所有节点组成的分割向量  $\mathbf{O} = \mathbf{I}_{\text{NC}}, \mathbf{I}_{\text{NC}}$  中所有分量都为 1, NC 为有限元节点自由度数目.

2.2 分区异步长显式并行求解

显式异步长求解过程需要重叠多层边界节点. 图 1 是以 2 倍步长比的网格划分示意图. 圆形表示内部节点, 用 I 表示; 三角形节点表示边界节点, 用 B 表示; 正方形节点表示外部节点, 用 E 表示. 阴影部分为共享单元. 由节点重叠和分割单元相邻分区共享可知: 分区边界节点是相邻分区的外部节点.

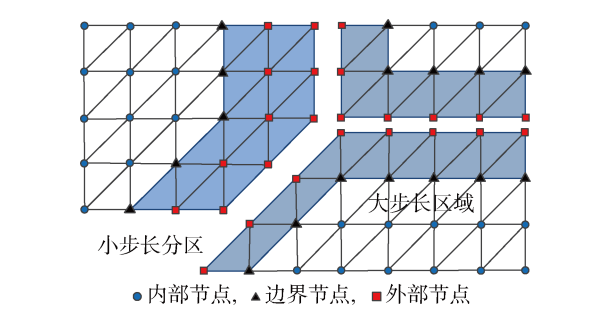


图 1 异步长计算多重节点划分示意  
Fig. 1 Mesh partition for asynchronous time step

参照分区节点分割向量的定义式, 小步长分区分割向量为

$$\mathbf{O}_i = \begin{bmatrix} 1 & \cdots & 1 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & \cdots & 0 & 1 & 1 \end{bmatrix} \quad (9)$$

式中:  $\mathbf{O}_i$  代表内部节点分割向量 (包含边界节点);  $\mathbf{O}_E$  为外部节点分割向量. 对步长比为  $m_i$  的分区 (为简化公式暂不包含阻尼), 离散的结构动力学方程可以写为

$$\mathbf{M}_i \ddot{\mathbf{u}}_i^{n+\frac{j}{m_i}} + \mathbf{K}_i \mathbf{u}_i^{n+\frac{j}{m_i}} = \mathbf{f}_i^{n+\frac{j}{m_i}} \quad (10)$$

式中:  $\mathbf{M}_i$  与  $\mathbf{K}_i$  为第  $i$  个分区的质量矩阵与刚度矩阵;  $\mathbf{f}_i^{n+\frac{j}{m_i}}$  表示第  $i$  个分区的外部节点力.

对于小步长分区的一个子循环时间步  $j$ , 内部

节点与边界节点的加速度可以写为

$$\begin{aligned} \widetilde{\mathbf{M}}_i \ddot{\mathbf{u}}_1^{n+\frac{j}{m_i}} = & \mathbf{O}_i^T \mathbf{f}_i^{n+\frac{j}{m_i}} - \widetilde{\mathbf{K}}_i \mathbf{u}_1^{n+\frac{j}{m_i}} - \mathbf{O}_i^T \mathbf{K}_i \mathbf{O}_E \mathbf{u}_E^{n+\frac{j}{m_i}} \end{aligned} \quad (11)$$

式中:  $\widetilde{\mathbf{M}}_i = \mathbf{O}_i^T \mathbf{M}_i \mathbf{O}_i$  和  $\widetilde{\mathbf{K}}_i = \mathbf{O}_i^T \mathbf{K}_i \mathbf{O}_i$  表示分区质量矩阵、刚度矩阵按布尔分割向量生成的分块矩阵;  $\mathbf{u}_E^{n+\frac{j}{m_i}}$  表示外部节点的位移预测值. 在子循环时间步内, 外部节点的加速度计算公式为

$$\begin{aligned} \widetilde{\mathbf{M}}_E \ddot{\mathbf{u}}_E^{n+\frac{j}{m_i}} = & \mathbf{O}_E^T \mathbf{f}_E^{n+\frac{j}{m_i}} - \widetilde{\mathbf{K}}_E \mathbf{u}_E^{n+\frac{j}{m_i}} - \mathbf{O}_E^T \mathbf{K}_i \mathbf{O}_i \mathbf{u}_B^{n+\frac{j}{m_i}} \end{aligned} \quad (12)$$

式中:  $\widetilde{\mathbf{M}}_E = \mathbf{O}_E^T \mathbf{M}_E \mathbf{O}_E$  和  $\widetilde{\mathbf{K}}_E = \mathbf{O}_E^T \mathbf{K}_i \mathbf{O}_i$  为与外部节点自由度有关的分块质量与刚度矩阵. 外部节点需要传递相邻分区边界节点的位移预测值  $\mathbf{u}_B^{n+1}$ , 子循环边界节点信息  $\mathbf{u}_B^{n+\frac{j}{m_i}}$  在显式计算过程中可以根据时间步传递. 对于小步长分区第二子循环步 (2 倍步长为例), 可按照式 (11) 求得分区内部节点以及边界节点的加速度.

对于大步长分区, 内部与边界节点的加速度求解公式与式 (10) 和 (11) 相似, 不同的是此时分区时间步长为  $\Delta t_i = m_i \Delta t_r$ , 预测步与校正步中求解公式需要将时间步长换成  $m_i \Delta t_r$ . 大步长分区质量矩阵与刚度矩阵需按照分区分割向量分块.

3 数值算例分析

3.1 并行计算性能参数

算法的有效性在国家超级计算广州中心的“天河二号”超级计算机上进行测试, 文中算例均采用 CPU 核心计算. 并行算法的性能常用加速比与并行计算效率来进行衡量. 异步长并行算法中, 结构分区数以及步长比不同都会导致计算加速比不同. 定义结构分区数目  $k$  固定, 步长比  $m$  不同时加速比为

$$S_{ki} = t_{km} / t_{ki} \quad (13)$$

定义  $m$  固定而  $k$  不同时加速比为

$$S_{mi} = t_{mk} / t_{mi} \quad (14)$$

计算效率为衡量一个处理器的计算能力被有效利用的百分比, 定义为

$$E = S / p \quad (15)$$

式中:  $S$  为加速比;  $p$  为使用计算核心数目.

3.2 弹性杆承受集中轴向荷载

图 2 为弹性杆在轴向集中荷载作用下结构示意图. 杆左端固定, 右端点 A 受轴向集中载荷. 杆材料密度  $\rho$  为 1 000 kg/m<sup>3</sup>, 弹性模量 1 GPa, 横截面积 0.001 m<sup>2</sup>, 杆的长度  $l$  为 1 m, 自由端加载外力  $f_{\text{ext}}$  为 1 000 N.  $x$  轴代表杆轴向坐标. 整个结构划分为 4 个子分区, 每个子分区长度为 0.25 m, 每个子分区离

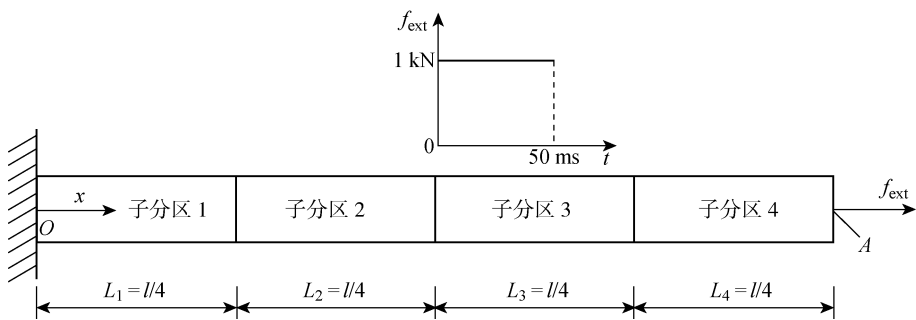


图 2 弹性杆受轴向集中载荷

Fig. 2 The elastic rod subjected to axial concentrated load

散为 200 个两节点杆单元,整个结构有 800 个单元, 801 个节点. 仿真时间 0.05 s.

子分区 1、2、3 采用相同时间步长,步长  $\Delta t$  为  $2.5 \mu\text{s}$ . 子分区 4 依次使用相同时间步长  $\Delta t_r$  ( $\Delta t = \Delta t_r$ )、5 倍时间步长 ( $\Delta t = 5\Delta t_r$ ) 与 10 倍时间步长 ( $\Delta t = 10\Delta t_r$ ) 共 3 种情况.

图 3 为子分区 4 采用不同时间步长比,杆右端点 A 轴向位移( $D$ )曲线与理论位移曲线. 图 4 为图 3 的局部放大图. 图 3 和 4 表明,随着分区时间步长

比的增加,位移的计算精度没有出现明显下降,依旧保持较高精度.

为比较网格特性对计算精度的影响,将子分区 4 进行细分离散,离散 2 倍,此时含有 400 个单元,而子分区 1~3 维持 200 个单元不变. 子分区 1~3 仍采用步长  $\Delta t$  为  $2.5 \mu\text{s}$ ,子分区 4 采用步长比为 10 的异步长方法,计算杆右端点 A 的轴向位移. Belyschko 方法是基于多重节点计算的经典方法,点 A 位移与 Belyschko 方法<sup>[9]</sup>进行比较.

采用 2 种方法计算得到的右端点位移曲线如图 5 所示,10 倍步长比下位移曲线局部放大如图 6 所示.

为了定量分析不同方法的计算精度,定义位移最大误差  $D_{\max}^e$  与位移误差和  $D_{\text{sum}}^e$  两个计算指标.

$$\left. \begin{aligned} D_{\max}^e &= |D_{\text{the}} - D_{\text{pro}}|_{\max} \\ D_{\text{sum}}^e &= \sum_{i=1}^n |D_{\text{the}} - D_{\text{pro}}| \end{aligned} \right\} \quad (16)$$

式中:  $n$  表示主时间步数;  $D_{\text{the}}$  为节点 A 理论位移;  $D_{\text{pro}}$  为采用给定方法计算的节点位移. 统计的节点位移精度指标见表 1 所示.

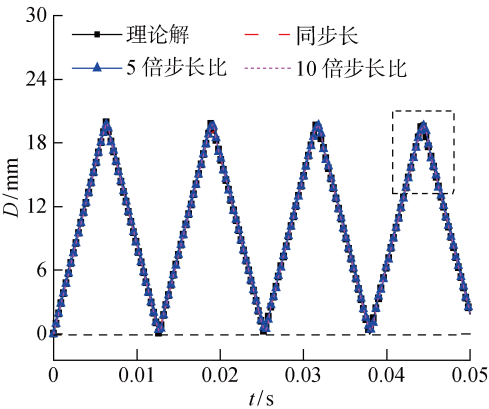


图 3 右端点 A 轴向位移曲线

Fig. 3 Axial displacement curves of point A

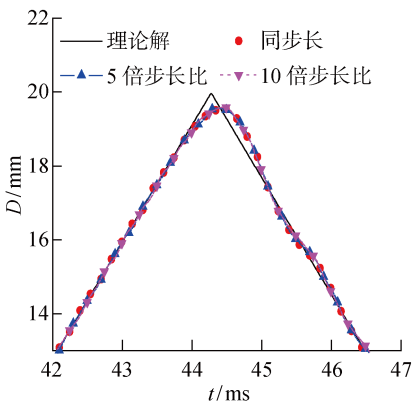


图 4 图 3 的局部放大

Fig. 4 Local amplification of Fig. 3

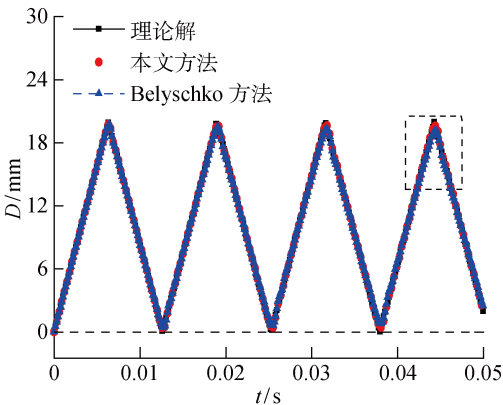


图 5 采用 10 倍步长比时不同方法右端点 A 位移曲线  
Fig. 5 Displacement curves of point A for different methods under  $m=10$

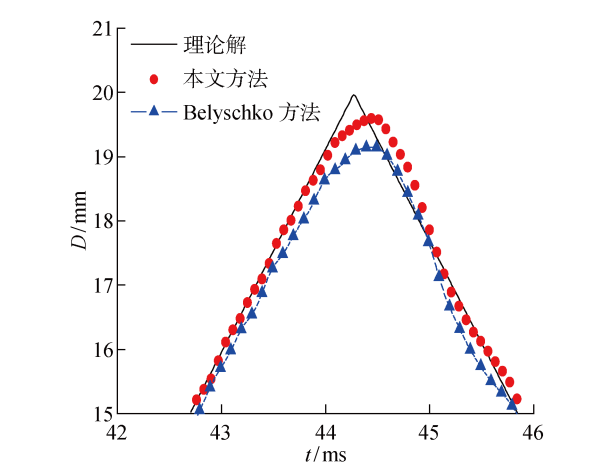


图 6 图 5 的局部放大  
Fig. 6 Local amplification of Fig. 5

表 1 节点 A 位移计算精度指标				
Table 1 Calculation index of nodal A displacement accuracy				
m	本文方法		Belyschko 方法	
	$D_{\max}^e/\text{mm}$	$D_{\text{sum}}^e/\text{mm}$	$D_{\max}^e/\text{mm}$	$D_{\text{sum}}^e/\text{mm}$
1	0.502	135	0.627	477
5	0.716	215	0.891	681
10	0.824	382	1.036	843

从表 1 中可见:采用相同计算模型,随着步长比的增加,2 种算法的最大计算误差以及误差和都在逐渐增加,计算精度相应降低.在相同步长比的条件下,与 Belyschko 方法相比,本文方法的最大误差以及误差和在数值上更小,计算指标变化的幅度也较小,表明本文方法具有较高的计算精度与稳定性.

3.3 多层建筑受冲击载荷

多层钢结构建筑模型如图 7 所示,整体建筑高 39.5 m,楼层底面长度为 48 m,宽为 16 m,建筑共有 13 层,其中最低层为高 3.5 m,其他楼层为 3 m.整个建筑结构采用 8 节点六面体单元离散,离散后的该建筑模型含有 6 918 660 节点,5 212 768 单元,20 755 980 自由度.仿真工况计算采用对结构左侧面的右三角脉冲冲击载荷.动力响应分析总时间为 0.6 s,大步长分区时间步长  $\Delta t=200\text{ ms}$ ,总计 30 000 个主计算时间步.

为了验证本文并行算法的计算效率,建筑结构最右端(图 7 中虚线框区域)接受冲击载荷部位子分区采用较小的时间步长计算,计算建筑高层点 B 的位移曲线, $P$  为冲击载荷, $P_{\max}$  为三角冲击载荷的最大值.将整个建筑单元分别划分为 120、240、480 与 960 个子区域,小步长分区网格在 120 子分区时占据 6 个分区.

图 8 为建筑结构在步长比为 4 的情况下节点 B 的位移时间曲线,该实例计算结果与 LS-DYNA 软件计算结果较相符.建筑结构节点位移没有理论解,位移求解精度只做定性比较.

异步长并行算法中,分区数目与分区步长比都会影响并行计算效率.表 2 为各个分区采用相同的时间时根据显式异步长算法在不同计算核数下得到的并行效率.从表 2 中可知,随着计算核数的增加,计算总时间逐渐减少,计算加速比增加,但是从计算效率的角度来看,并行计算效率在逐渐下降.

表 3 为在图 8 中小步长分区  $m=4$  的条件下,

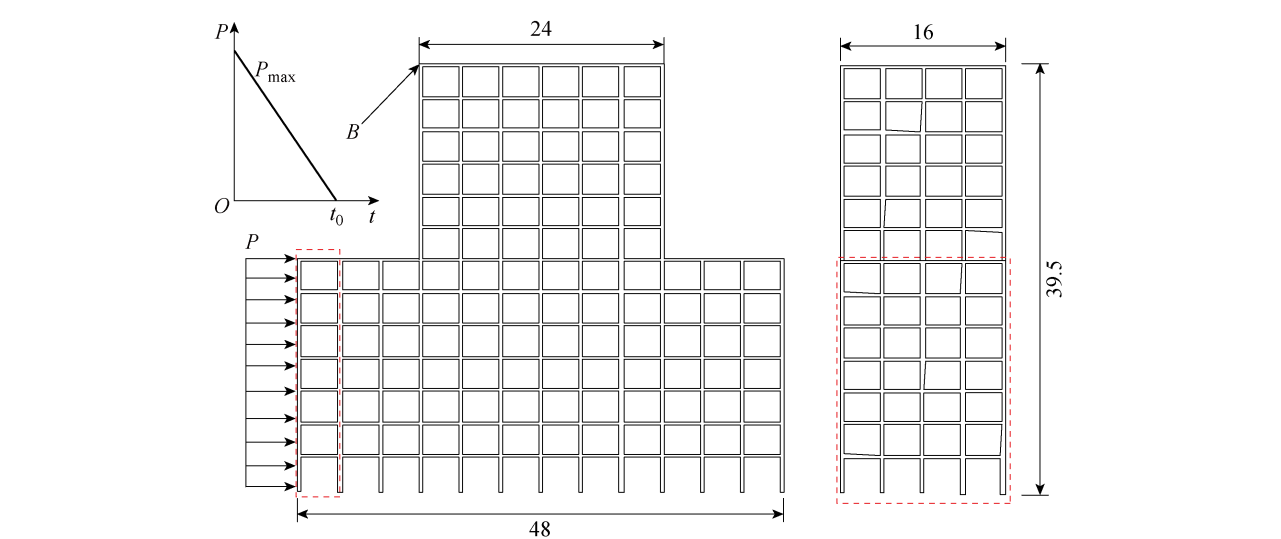


图 7 多层建筑受冲击载荷(m)  
Fig. 7 Multi-story building subjected to impact load (m)

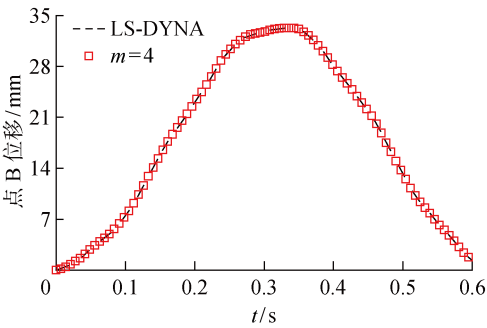


图 8 多层建筑节点 B 的位移曲线

Fig. 8 Displacement curves of point B for multi-storey building

表 2 分区同步长条件下计算效率 (m = 1)

Tab. 2 Parallel speedup and efficiency for different CPU numbers (m = 1)

CPU 核数	计算时间/s	加速比	并行效率/%
120	16 394	1	100
240	9 065	1.81	90.5
480	5 229	3.13	78.3
960	3 128	5.24	65.5

表 3 同步长比条件下计算效率 (m = 4)

Tab. 3 Parallel speedup and efficiency with the same step ratio (m = 4)

CPU 核数	计算时间/s	加速比	并行效率/%
120	9 281	1	100
240	5 185	1.79	89.7
480	3 024	3.07	76.8
960	1 792	5.18	64.8

随着分区数目变化的显式异步长算法并行效率. 表 3 所得并行计算效率结论与表 2 类似, 由两表格数据相比较可知: 异步长并行条件计算下, 步长比的选取对于并行效率没有实质性提升; 但是, 局部区域采用较小的时间步长, 其他区域采用较大时间步长, 计算总时间大幅度减少.

表 4 为在核数等于 480 时, 随着最右端分区步长比变化所得的并行计算效率. 由表 4 可以看出, 随着步长比的增加, 计算总时间与计算时间比率大幅下降, 计算时间比率减少幅度在逐渐下降. 步长比增加时, 大步长分区计算时间降低有限, 在一个异步长通信周期内, 分区负载不均衡现象显现, 因此研究负载均衡算法可以进一步提升显式异步算法性能, 降低计算时间比率.

表 4 相同分区数不同步长比下计算时间

Tab. 4 Computation time with different step ratios under the same partition number

步长比	计算时间/s	时间比率/%
1	5 229	100
2	3 908	74.6
4	3 024	57.8
8	2 551	48.7
16	2 103	40.2

4 结语

本文提出一种基于多重节点的预测校正形式显式异步结构动力学并行计算方法, 该方法具有如下特点:

- (1) 采用显式 Newmark 求解格式, 对时间离散形式的结构动力学方程自由度解耦, 边界信息传递只与相邻分区有关, 提高了并行计算过程信息传递效率.
  - (2) 采用节点分割将计算模型分割成若个子区域, 各个子区域选择各自的时间步长, 解决了因局部精细网格造成的临界时间步长增加这一难题.
  - (3) 采用多重节点的分区耦合策略, 相邻分区的部分区域设定为耦合分区, 子循环过程中, 预测波形得以完整地在整个耦合区域传播, 而不是传统方法中经过截断的波形. 这在一定程度上提高了计算精度.
- 研究结果也进一步表明, 随着步长比增加, 分区负载不平衡在一定程度上阻碍了计算效率的提升. 因此, 下一步研究的重点是提高异步并行负载的均衡性能.

参考文献:

[1] 王小庆, 金先龙, 曹源. 大规模输水隧道水锤效应三维数值模拟[J]. 上海交通大学学报, 2016, 50(1): 98-102.  
WANG Xiaoqing, JIN Xianlong, CAO Yuan. Three-dimensional simulation of water hammer of large-scale water conveyance tunnel[J]. Journal of Shanghai Jiao Tong University, 2016, 50(1): 98-102.

[2] 张伟伟, 金先龙. 统一格式的显式与隐式任意混合异步算法[J]. 力学学报, 2013, 46(3): 436-446.  
ZHANG Weiwei, JIN Xianlong. An arbitrarily mixed explicit-implicit asynchronous integration algorithm based on uniform discretization format[J]. Chinese Journal of Theoretical and Applied Mechanics, 2013, 46(3): 436-446.

- [3] GHANEM A, TORKHANI M, MAHJOUBI N, *et al.* Arlequin framework for multi-model, multi-time scale and heterogeneous time integrators for structural transient dynamics[J]. **Computer Methods in Applied Mechanics and Engineering**, 2013, 254: 292-308.
- [4] 张友良, 冯夏庭, 茹忠亮. 基于区域分解算法的岩土大规模高性能并行有限元系统研究[J]. **岩石力学与工程学报**, 2004, 23(21): 3636-3641.  
ZHANG Youliang, FENG Xiating, RU Zhongliang. Large-scale high performance parallel finite element system based on domain decomposition method in geomechanics[J]. **Chinese Journal of Rock Mechanics and Engineering**, 2004, 23(21): 3636-3641.
- [5] FARHAT C, ROUX F X. A method of finite element tearing and interconnecting and its parallel solution algorithm[J]. **International Journal for Numerical Methods in Engineering**, 1991, 32(6): 1205-1227.
- [6] 刘钺, 叶子晟, 胡海岩. 基于区域分解的柔性多体系统高效并行算法[J]. **中国科学: 物理学 力学 天文学**, 2017, 47(10): 104603.  
LIU Cheng, YE Zisheng, HU Haiyan. An efficient parallel algorithm for flexible multibody systems based on domain decomposition method[J]. **Scientia Sinica Physica, Mechanica & Astronomica**, 2017, 47(10): 104603.
- [7] KOZUBEK T, VONDRAK V, MENSİK M, *et al.* Total FETI domain decomposition method and its massively parallel implementation[J]. **Advances in Engineering Software**, 2013, 60/61: 14-22.
- [8] SMOLINSKI P. An explicit multi-time step integration method for second order equations[J]. **Computer Methods in Applied Mechanics and Engineering**, 1992, 94(1): 25-34.
- [9] BELYSCHKO T, LU Y Y. Explicit multi-time step integration for first and second order finite element semidiscretizations[J]. **Computer Methods in Applied Mechanics and Engineering**, 1993, 108(3/4): 353-383.
- [10] PRAKASH A, TACIROGLU E, HJELMSTAD K D. Computationally efficient multi-time-step method for partitioned time integration of highly nonlinear structural dynamics[J]. **Computers and Structures**, 2014, 133: 51-63.
- [11] 刘占生, 张云峰. 非线性壁板颤振计算的子循环预测校正方法研究[J]. **航空动力学报**, 2007, 22(5): 761-767.  
LIU Zhansheng, ZHANG Yunfeng. Nonlinear panel flutter computation with subcycling and predict correct scheme[J]. **Journal of Aerospace Power**, 2007, 22(5): 761-767.
- [12] 高双武, 强洪夫, 周伟, 等. 基于子循环-预测校正的三维 SRM 点火瞬态流固耦合数值模拟[J]. **计算力学学报**, 2011, 28(Sup.): 84-89.  
GAO Shuangwu, QIANG Hongfu, ZHOU Wei, *et al.* Fluid-structure interaction numerical simulation of 3D SRM during ignition with subcycling based predict-correct scheme[J]. **Chinese Journal of Computational Mechanics**, 2011, 28(Sup.): 84-89.
- [13] KARIMIS S, NAKSHATRALA K B. On multi-time-step monolithic coupling algorithms for elastodynamics[J]. **Journal of Computational Physics**, 2014, 273: 671-705.
- [14] SUBBER W, MATOUS K. Asynchronous space-time domain decomposition method with localized uncertainty quantification[J]. **Computer Methods in Applied Mechanics and Engineering**, 2017, 325: 369-394.
- [15] HUGHES T J R. The finite element method: Linear static and dynamic finite element analysis[M]. New Jersey, USA: Prentice Hall, 2012.

(本文编辑: 蒋霞)