



差分法による 弾性波動並列シミュレーション

東京大学 情報学環 / 地震研究所 森太志 f-mori@eri.u-tokyo.ac.jp

東京大学 情報学環 / 地震研究所 古村 孝志

本日の目次

- ・はじめに
 - 有限差分法(FDM)による弾性波動計算の概要
 - ppOpen-APPL/FDM の概要
 - FX10によるパフォーマンステスト
 - 応用例: 大規模連成計算

- ppOpen-APPL/FDMの演習
 - 利用方法
 - 演習

FDMによる弾性波動計算

• 構成方程式

- 弾性体の釣り合いの式(運動方程式)

$$\rho \ddot{u} = \frac{\partial \sigma_{xp}}{\partial x} + \frac{\partial \sigma_{yp}}{\partial y} + \frac{\partial \sigma_{zp}}{\partial z} + f_p, (p = x, y, z)$$
(1)
$$\ddot{u}: \mathbf{n} 速度, \sigma: \mathbf{c} \mathbf{D}, \rho: 密度, f: 外 \mathbf{D}$$

- 等方完全弾性体の応力

 $\sigma_{pq} = \lambda (e_{xx} + e_{yy} + e_{zz}) \delta_{pq} + 2\mu e_{pq}, (p, q = x, y, z)$ (2) λ, μ : Lame定数、 δ : クロネッカのデルタ

• 歪みは変位の空間微分で求められる $e_{pq} = \frac{1}{2} \left(\frac{\partial u_p}{\partial q} + \frac{\partial u_q}{\partial p} \right), \quad (p,q=x,y,z)$ (3)

e: 歪み

弾性波動の陽的計算

時間発展により波動伝播を進めるために、式(1)
 の速度変数を中間変数とする

$$\dot{u}_{p}^{n+\frac{1}{2}} = \dot{u}_{p}^{n-\frac{1}{2}} + \frac{1}{\rho} \left(\frac{\partial \sigma_{xp}^{n}}{\partial x} + \frac{\partial \sigma_{yp}^{n}}{\partial y} + \frac{\partial \sigma_{zp}^{n}}{\partial z} + f_{p}^{n} \right) \Delta t, \ (p = x, y, z) \qquad (4)$$
$$\dot{u}^{n+1/2}{}_{p} \ \hbar \Xi$$

 ・式(2)と式(3)を結合した式(5)を用いて中央差分に 基づく時間積分

$$\sigma_{pq}^{n+1} = \sigma_{pq}^{n} + \left[\lambda \left(\frac{\partial \dot{u}_{x}^{n+\frac{1}{2}}}{\partial x} + \frac{\partial \dot{u}_{y}^{n+\frac{1}{2}}}{\partial y} + \frac{\partial \dot{u}_{z}^{n+\frac{1}{2}}}{\partial z}\right) \delta_{pq} + \mu \left(\frac{\partial \dot{u}_{p}^{n+\frac{1}{2}}}{\partial q} + \frac{\partial \dot{u}_{q}^{n+\frac{1}{2}}}{\partial p}\right)\right] \Delta t, \ (p,q) = (x, y, z) \quad (5)$$

格子モデル



- A) 変位や応力、物性値など全ての変数を同一格子点上に配置する一般 格子
- B) 変位を半格子ずれた位置に定義するスタガード格子
 - 変数の微分が定義される位置に関連の変数が位置するため計算精度が良い

FDMによる弾性波動計算

FDMによる式(4)と(5)の空間微分
 –中心差分による計算(2次精度、4次精度、8次精度)

(2次精度)
$$\frac{d}{dx}\sigma_{pq}(x,y,z) \simeq \frac{1}{\Delta x} \left[\sigma_{pq}\left(x + \frac{\Delta x}{2}, y, z\right) - \sigma_{pq}\left(x - \frac{\Delta x}{2}, y, z\right) \right]$$

(4次精度) $\frac{d}{dx}\sigma_{pq}(x,y,z) \simeq \frac{1}{\Delta x} \left[\frac{9}{8} \left\{ \sigma_{pq}\left(x + \frac{\Delta x}{2}, y, z\right) - \sigma_{pq}\left(x - \frac{\Delta x}{2}, y, z\right) \right\} - \frac{1}{24} \left\{ \sigma_{pq}\left(x + \frac{3\Delta x}{2}, y, z\right) - \sigma_{pq}\left(x - \frac{3\Delta x}{2}, y, z\right) \right\} \right]$

(8次精度)
$$\frac{d}{dx}\sigma_{pq}(x,y,z) \simeq \frac{1}{\Delta x} \left[\frac{1225}{1024} \left\{ \sigma_{pq} \left(x + \frac{\Delta x}{2}, y, z \right) - \sigma_{pq} \left(x - \frac{\Delta x}{2}, y, z \right) \right\} - \frac{245}{3072} \left\{ \sigma_{pq} \left(x + \frac{3\Delta x}{2}, y, z \right) - \sigma_{pq} \left(x - \frac{3\Delta x}{2}, y, z \right) \right\} + \frac{49}{5120} \left\{ \sigma_{pq} \left(x + \frac{5\Delta x}{2}, y, z \right) - \sigma_{pq} \left(x - \frac{5\Delta x}{2}, y, z \right) \right\} - \frac{5}{7168} \left\{ \sigma_{pq} \left(x + \frac{7\Delta x}{2}, y, z \right) - \sigma_{pq} \left(x - \frac{7\Delta x}{2}, y, z \right) \right\} \right]$$

ppOpen-APPL/FDM の概要

ppOpen-APPL/FDM 概要

- 1. 弾性波動並列シミュレーション(地震)
 - Staggered グリッド、 陽解法
 - 3次元/2次元モデル
 - 等間隔格子
 - 微分: 2次、4次、8次精度
 - MPI並列は3次元領域分割
 - MPI/OpenMPハイブリッド並列



3次元領域分割 (赤文字:MPIランク)

ppOpen-APPL/FDM 概要 (cont.)

- 2. サンプルプログラムとインターフェイス
 - 観測点の波形

9

- 波動場のスナップショット



ppOpen-APPL/FDM概要 (cont.)

- ・ドキュメント
 - ppOpen-APPL/FDMの使い方:ユーザマニュアル
 コード内のモジュール説明

ontents- Ontice of semine, 2D semine, 3D		_
Outline of setimic_3D 5 11 Paralid immits of setimic wave propagation in heterogeneous clastic molia using ppOperFind 5 12 Grid and coordinate system 5 13 Grid and coordinate system 5 14 Boundary condition 6 14 A blocting boundary 6 14 A blocting boundary 7 15 Adatatist attemations 7 16 Spatial Differentiation 8 17 Input parameters 8 18 Requirements of time integration (CFL Condition) 9 19 Output and vinalization of seimic waves 9 10 Output and vinalization of seimic waves 10 11 Deather partiterioning and MPI 10 12 Particle pDS multinitions and performance 10 12 Deather partiterioning furture 11 13 Performance of particle PDM simulation. 11 14 Fired PM multinition and performance 14 13 I model polePDM, convert 17 14 The polePDM multinition. 12 15 simula gave 15 16 addite polePDM, convert 17 17 addite polePDM, convert 17 18 amodite polePDM, multiniti 22	ontents.	
1.1 Parallel simulation of semic wave propagation in heterogeneous elastic media using poperEVM. 55 1.2 Grid and coordinate system. 56 1.3 Equations of Medico for JD Semic Wavefields. 56 1.4 Roberbing boundary. 66 1.4.1 Absorbing boundary. 77 1.5 Areatics: attenuations. 77 1.6 Apathal Differentiation. 78 1.6 Apathal Differentiation. 76 1.7 Ioput parameters 78 1.8 Requirements of time integration (CFL Condition). 96 1.9 Domain partitioning and MPI. 100 2.1 Domain partitioning and MPI. 101 2.1 Parallel programmeters 14 3.1 Requirements of time integration (CFL Condition). 102 3.2 Parallel programming mucture. 102 3.1 Domain partitioning and MPI. 102 3.2 Parallel programming mucture. 11 3.1 Parallel programming mucture. 12 3.1 Standard pays 15 3.1 module polePDM_m_network 17 3.1 module polePDM_m_convet 17 3.1 module polePDM_m_convet. 12 3.1 module polePDM_m_convet. 12 3.1 module	Outline of seismic_2D/seismic_3D	5.
pppperIDM 5 12 Gotal act coordinate system 5 13 Equations of Motion for 3D Seimic Wavefields 5 14 Bondary conditions 6 14 Al Motering boundary 6 14 Al Motering boundary 7 15 Adatatic attemations 7 16 Spatial Differentiation 7 16 Spatial Differentiation 7 17 Dept parameters 8 18 Requirements of time integration (CEL Condition) 8 19 Output and visualization of stemics waves 9 10 Output and visualization of stemics waves 10 11 Doutput parameters 11 21 Doutput and visualization of stemics waves 11 21 Doutput and visualization of attrimente 12 31 In model postFDM	1.1 Parallel simulation of seismic wave propagation in heterogeneous elastic media usi	ing
12 Gord and coordinate system. 5 13 Equations of Motion for 1D Semic Wavefields 5 14 Boundary conditions. 6 14 Boundary conditions. 6 14 Aloreting boundary. 6 15 Arelatic attenuitons 7 15 Arelatic attenuitons 7 15 Arelatic attenuitons 7 16 Spatial Differentiation of semic waves 8 17 Input parameters 8 18 Requirements of time inegration (CFL Condition) 9 19 Output and visualization of semic waves 9 10 Output and visualization of semic waves 9 12 Parallel programming utwave 10 12 Parallel programming utwave 10 12 Parallel programming utwave 10 13 Personal of the MPI Hybrid model 12 14 File MPI models to thered MPI Hybrid model 12 15 restandlar postPDM_m_kenetb 17 16 andiale postPDM_m_winetb 17 13 model postPDM_m_convext 18 14 and ondiale postPDM_m_postPM_m_motion 22 15 module postPDM_m_postPM_m_motion 23 16 module postPDM_m_postPM_m_motion 24 <	ppOpenFDM	5.
1.3 Equations of Motion for 3D Seimic Wavefields 5 1.4 Bomdary conditions 6 1.4.1 Absorbing boundary. 6 1.4.1 Absorbing boundary. 7 1.5 Analatis attenuotos 7 1.6 Spatial Differentiation 7 1.6 Spatial Differentiation 8 1.7 Diput parameters 8 1.8 Requirements of time integration (CFL Condition) 9 1.9 Output and visualization of seimic waves 9 9 Parallel FDM simulations of seimic waves 9 9 Parallel FDM simulations 10 1.1 Docama particleing and MPI 10 2.1 Paralle programming structure 11 3.1 Parallel programming structure 11 3.1 Parallel programming structure 12 3.1 Parallel programming structure 11 3.1 Parallel programming structure 12 3.1 Structure 12 3.1 Structure 14 3.1 simulation and parformation 12 3.1 Structure 13 3.1 Structure 14 3.1 Structure 17 3.1 Structure 17 3.1 Structur	1.2 Grid and coordinate system.	5.
1.4 Boundary continuent. 8 1.4 A lostering boundary. 6 1.4 A loster boundary. 7 1.5 Analatic attenuations. 7 1.5 Analatic attenuations. 7 1.6 Spinial Differentiation. 8 1.7 Input parameters 8 1.8 Requirements of time integration (CFL Condition) 9 1.9 Output and vinalization of seimic waves 9 1.9 Departual CFM initiations and performance. 10 2.1 Densing partitioning and MPI. 10 2.3 Performance of parallel FDM initiation. 11 2.4 File MPI models vs. freed MPI Hybrid model. 12 2.4 File MPI models vs. freed MPI Hybrid model. 12 3.1 simular polysimular verser. 14 3.1 model polysiPDM_m_sternet. 16 3.1 model polysiPDM_m_sternet. 16 3.1 model polysiPDM_m_nutput 22 3.1 model polysiPDM_m_nutput 23 3.1 model polysiPDM_m_nutput 24 3.1 model polysiPDM_m_nutput 23 3.1 model polysiPDM_m_nutput 24 3.1 model polysiPDM_m_nutput 25 3.1 model polysiPDM_m_nutput 25	1.3 Equations of Motion for 3D Seismic Wavefields	5.
1.4.1 Asserting boundary	1.4 Boundary conditions	6.
1.4 2 Free surface boundary 7 1.5 Analatist structures 7 1.6 Spatial Differentiation 7 1.6 Spatial Differentiation 8 1.7 Input parameters 8 1.8 Requirements of time singuration (CFL Condition) 9 1.9 Output and visualization of seimic waves 9 1.9 Design and visualization of seimic waves 9 2.1 Densing participations and performance 10 2.1 Densing participations and performance 10 2.1 Densing participations and performance 11 2.2 Paralicle pograming structures 11 2.4 Fill MPI model vs. thread MPI Hybrid model. 12 2.4 Fill MPI model vs. thread MPI Hybrid model. 12 3.1 stemidal gav 15 3.1 a model popherDM_m_patters 17 3.1 a model popherDM_m_patters 19 3.1 a model popherDM_m_patters 22 3.1 a model popherDM_m_patters 24 3.1 model popherDM_m_patters 24 3.1 model popherDM_m_patters 26 3.1 model popherDM_m_patters 27 3.1 model popherDM_m_patters 26 3.1 model popherDM_m_patters	1.4.1 Absorbing boundary.	6
15 Andraits: attenuition: 7 15 Andraits: attenuition: 8 17 Input parameters: 8 18 Requirements of time inegration (CFL Condition) 9 19 Output and variantization of semine waves: 9 Parallel FDM immitation and performance: 10 2.1 Domain particularition of semine waves: 10 2.2 Parallel programming structure: 10 2.3 Parallel programming structure: 11 3.4 File MPI models: 12 4.7 File MPI models: 12 1.3 reducting and MPI 100 2.1 seminated to:st iterad MPI Hyster model 12 1.4 File MPI models: 12 1.5 and the postPDM_m_scheetb 17 3.1 module postPDM_m_convert 16 3.1 module postPDM_m_convert 19 3.1 module postPDM_m_convert 23 3.1 module postPDM_m_convert 24 3.1 module postPDM_m_convert 25 3.1 module postPDM_m_convert 25 3.1 module postPDM_m_convert 25 3.1 module postPDM_m_convert 26 3.1 module postPDM_m_convert 27 3.1 module postPDM_m_con	1.4.2 Free surface boundary	7.
16 Spatial Differentiation 8 17 Suppl parameters 8 18 Requirements of time integration (CFL Condition) 9 19 Output and visualization of stemmic waves 9 19 Output and visualization of stemmic waves 9 21 Denning partitions and performance 10 21 Denning partitions and performance 10 21 Denning partitions and performance 11 21 Denning partitions and performance 11 21 Partition perparaming structure 11 23 Performance of partitle FPM simulation 11 24 Fut MPI model vs. thread MPI Hybrid model. 12 24 ferences 14 31 simulal pay 15 31 I model repoleFDM m_partition. 17 31 a model poleFDM m_partition. 12 31 A module poleFDM m_partition. 12 31 A module poleFDM m_partition. 22 31 a module poleFDM m_partition. 24 31 A module poleFDM m_partition. 26 31 A module poleFDM m_partition. 27 31 A module poleFDM m_partition. 26 31 A module poleFDM m_partitin.	1.5 Anelastic attenuations	7.
1.7 Input parameters 8 1.8 Requirements visualization of steimic waves 9 1.8 Requirements of steimic waves 9 1.9 Output and Visualization of steimic waves 9 2.10 main partitions and performance 10 2.1 Domain partitioning and MPI 10 2.2 Parallel FDM simulation 10 2.3 Performance of availal FDM simulation. 11 2.4 Flat MPI model vs. freed MPI Hybrid model. 12 1.5 Issued approx 15 3.1 simuld approx 15 3.1 module pohFDM	1.6 Spatial Differentiation-	8.
1.8 Requirements of time integration (CFL Condition) 9 1.9 Output and visualization of string waves 0.0 Parallel FDM simulation of string waves 0.0 1.1 Domain participing and MP1 10.0 1.2 Density and yaves 10 2.1 Density participing and MP1 10.0 2.1 Density participing and MP1 10.0 3.2 Participing maining structures 11.1 3.1 Participing maining structures 12.0 A Fulk DP1 model via. thread API Hybrid model. 12.0 1.1 model participing participing thread t	1.7 Input parameters	8.
10 Output and visualization of relamine waves 9. 10 Parallel FDM simulation and performance. 10. 11 Domain patholing and MP1 10. 12.1 Parallel FDM simulation. 10. 12.1 Parallel FDM simulation. 11. 14.4 FM MP1 model vs. thread MP1 Hybrid model. 12. 15.1 Simulation postFDM_m_short 17. 1.1 Incidie postFDM_m_short 17. 1.1 model postFDM_m_short 17. 1.1 model postFDM_m_short 17. 1.1 model postFDM_m_short 17. 1.1 model postFDM_m_short 17. 1.2 modelie postFDM_m_short 19. 1.3 model postFDM_m_stort 19. 1.4 modelie postFDM_m_paration 24. 1.5 model postFDM_m_paration 24. 1.1 modelie postFDM_m_paration 24. 1.1 model postFDM_m_paration 24. 1.1 modelie postFDM_m_paration 26. 1.1 model postFDM_m_paration 30. 1.4 modelie postFDM_m_paration 30. 1.1 model postFDM_m_paration 30. 1.1 model postFDM_m_watch 37. 1.2	1.8 Requirements of time integration (CFL Condition)	9.
Parallel FDM simulation and performance. 10 2.1 Domain particle TDM simulation. 10 2.2 Parallel programming structure. 10 2.3 Performance of parallel FDM simulation. 11 3.1 Section Structure. 11 3.1 section Structure. 12 3.1 section Structure. 14 3.1 section Structure. 16 3.1 module postPDM m_betterb. 17 3.1 module postPDM m_perster. 26 3.1 module postPDM m_betterb. 27 3.1 module postPDM m_betterb. 37	1.9 Output and visualization of seismic waves	9-
1.1 Densing and MPI. 10. 2.1 Densing particular products and products an	Parallel FDM simulation and performance	10-
2.2 Parallel programming invurse. 11 2.2 Parallel rDM immiation. 11 2.4 Flat MPI model vs. thread MPI Hybrid model. 12. 2.4 Flat MPI model vs. thread MPI Hybrid model. 12. 1.4 Flat MPI model vs. thread MPI Hybrid model. 12. 1.5 issued approximation of the state st	2.1 Domain partitioning and MPI	10+
3.1 Performance of parallel FDM simulation. 11 3.2 Performance of parallel FDM simulation. 12 After MPI model vs. thread MPI Hybrid model. 12 4ferences 13 Module/subresutine reference 14 3.1 model repoleFDM_m_whereb 17 3.1 model repoleFDM_m_whereb 17 3.1 model repoleFDM_m_whereb 17 3.1 model repoleFDM_m_mentain 19 3.1 model repoleFDM_m_paramin 22 3.1 model repoleFDM_m_paramin 24 3.1 model repoleFDM_m_startec 27 3.1 model repoleFDM_m_startec 36 3.1 10 model repoleFDM_m_startec 37 3.1 11 model repoleFDM_m_startec 37 3.2 settmid14 39	2.2 Parallel programming structure.	11-
2.4 Fize MPR model vs. thread MPP Hybrid model. 12 detention of the second secon	2.3 Performance of parallel FDM simulation	11-
deferments 13 Module'subroutine reference 14 13 seimed.prv 15 14.1 module probeTDM_m_ketsetb 17 13.1 module probeTDM_m_centre 17 13.1 module probeTDM_m_centre 19 13.1 module probeTDM_m_centre 19 14.1 module probeTDM_m_centre 19 15.1 module probeTDM_m_contract 29 16.1 module probeTDM_m_contract 29 17.1 module probeTDM_m_contract 29 16.1 module probeTDM_m_contract 29 17.1 module probeTDM_m_contract 29 17.1 module probeTDM_m_contract 20 17.1 module probeTDM_m_contract 20 17.1 module probeTDM_m_contract 20 17.1 module probeTDM_m_contract 20 17.1 module probeTDM_m_contract 27 17.1 module probeTDM_m_contract 27 17.2 module probeTDM_m_contract 27 17.2 module probeTDM_m_contract 27 17.3 module probeTDM_m_contract 27 17.3 module probeTDM_m_contract 27 18.1 module probeTDM_m_contract 27	2.4 Flat MPI model vs. thread/MPI Hybrid model	12-
Module'unbruutine reference 14 3.1 mediate poskFDM_m_shortb 17 3.1.1 modiate poskFDM_m_conversa 18 3.1.3 module poskFDM_m_conversa 18 3.1.3 module poskFDM_m_conversa 19 3.1.4 module poskFDM_m_conversa 19 3.1.5 module poskFDM_m_output 22 3.1.6 module poskFDM_m_output 23 3.1.6 module poskFDM_m_output 25 3.1.6 module poskFDM_m_output 25 3.1.8 module poskFDM_m_output 25 3.1.8 module poskFDM_m_output 26 3.1.10 module poskFDM_m_withb 30 3.1.11 module poskFDM_m_withb 30 3.1.11 module poskFDM_m_withb 32 3.2 seim40A 36	ferences	13-
Modulevalue reference 14 Module probFDM_m_absorb 15 3.1 module probFDM_m_bench 17 3.1 module probFDM_m_bench 18 1.3 module probFDM_m_bench 19 3.1 module probFDM_m_bench 19 3.1 module probFDM_m_premit 28 3.1 module probFDM_m_premit 23 3.1 module probFDM_m_premit 24 3.1 module probFDM_m_premit 55 3.1 module probFDM_m_premit 26 3.1 module probFDM_m_premit 27 3.1 module probFDM_m_premit 30 3.1 module probFDM_m_mremit 30		
1.1 semid.pv 16 1.1 mode poleTDM_m_stort 17 1.1 mode poleTDM_m_convert 19 1.1 mode poleTDM_m_convert 19 1.1 mode poleTDM_m_convert 19 1.1 mode poleTDM_m_convert 19 1.1 mode poleTDM_m_convert 12 1.1 mode poleTDM_m_convert 13 1.1 mode poleTDM_m_prem 24 1.1 mode poleTDM_m_prem 24 1.1 mode poleTDM_m_prem 24 1.1 mode poleTDM_m_premt 25 1.1 mode poleTDM_m_mreft 30 1.1 mode poleTDM_m_mreft 37 1.1 mode poleTDM_m_mreft 36 1.1 mode poleTDM_m_mreft 36 1.1 mode poleTDM_m_mreft 37 1.1 mode poleTDM_m_mreft 39 1.1 mode poleTDM_m_mreft 39 1.1 mode poleTDM_m_mreft 39 1.1 mode poleTDM_m_mreft 39	Module/subroutine reference	14-
3.1.1 module ppokPDM_m_shorth 17 3.1.2 module ppokPDM_m_convar 18 3.1.3 module ppokPDM_m_shorth 19 3.1.4 module ppokPDM_m_output 22 3.1.5 module ppokPDM_m_output 23 3.1.5 module ppokPDM_m_output 23 3.1.6 module ppokPDM_m_output 23 3.1.7 module ppokPDM_m_output 23 3.1.8 module ppokPDM_m_output 25 3.1.8 module ppokPDM_m_output 25 3.1.8 module ppokPDM_m_output 26 3.1.1 module ppokPDM_m_output 30 3.1.10 module ppokPDM_m_watch 30 3.1.11 module ppokPDM_m_watch 37 3.2 setmidda 30	3.1 seism2d_psv	15-
1.1 model poleTDM convar. 18 1.3 model poleTDM 19 3.1 model poleTDM 19 3.1 model poleTDM 22 3.1 model poleTDM 23 3.1 model poleTDM 24 3.1 model poleTDM 24 3.1 model poleTDM	3.1.1 module ppohFDM_m_absorb	17-
3.1.3 module ppokFDM_m_textent 19 3.1.4 module ppokFDM_m_contract 22 3.1.5 module ppokFDM_m_contract 23 3.1.6 module ppokFDM_m_properties 24 3.1.7 module ppokFDM_m_properties 25 3.1.8 module ppokFDM_m_source 27 3.1.8 module ppokFDM_m_source 27 3.1.9 module ppokFDM_m_source 27 3.1.10 module ppokFDM_m_source 27 3.1.10 module ppokFDM_m_source 36 3.1.11 module ppokFDM_m_source 37 3.2 seimADa 30 3.2 immADa 30 3.1 21 module tpokFDM boundary. 50	3.1.2 module ppohFDM_m_comvar.	18-
1.4 module ppdrDM_m_medium 22 1.1 module ppdrDM_m_oparams 23 1.6 module ppdrDM_m_oparams 34 1.1 module ppdrDM_m_oparams 34 1.1 module ppdrDM_m_oparams 34 1.1 module ppdrDM_m_oparams 34 1.1 module ppdrDM_m_oparams 37 1.1 module ppdrDM_m_oparams 37 1.1 module ppdrDM_m_oparams 37 2.1 10 module ppdrDM_m_oparams 37 3.1 10 module ppdrDM_m_oparams 37 3.1 21 module ppdrDM_m_oparams 39 21 module ppdrDM_modulev_ 50	3.1.3 module ppohFDM_m_kernel.	19-
3.1 5 module poskFDM_m_output. 23 3.1 6 module poskFDM_m_params 24 3.1 7 module poskFDM_m_report. 25 3.1 8 module poskFDM_m_source 27 3.1 9 module poskFDM_m_with 30 3.1 10 module poskFDM_m_with 30 3.1 10 module poskFDM_m_with 36 3.1 11 module poskFDM_m_with 37 3.2 stam3dbs 30	3.1.4 module ppohFDM_m_medium	22.
3.1.6 module ppokPDM_m_preprint 24 3.1.7 module ppokPDM_m_report 55 3.1.8 module ppokPDM_m_route 77 3.1.9 module ppokPDM_m_routh 50 3.1.10 module ppokPDM_m_routh 30 3.1.11 module ppokPDM_m_wroth 32 3.1.11 module ppokPDM_m_wroth 77 3.1.11 module ppokPDM_m_wroth 70 3.1.21 module tpokPDM_m_wroth 30 3.2 seimAda 30	3.1.5 module ppohFDM_m_output.	23-
3.1 model postFDM_m_sequet. 15 3.1 model postFDM_m_store 27 3.1 model postFDM_m_store 77 3.1 model postFDM_m_starb 30 3.1 1 model postFDM_m_starb 36 3.1 1 model postFDM_m_starb 77 3.1 1 model postFDM_m_starb 77 3.1 1 model postFDM_m_starb 76 3.1 1 model postFDM_m_starb 77 3.1 1 model postFDM_m_starb 76 3.1 1 model postFDM_m_starb 50	3.1.6 module ppohFDM_m_params	24+
3.1 # module ppokFDM_m_source 27 3.1.9 module ppokFDM_m_mflib 000 3.1.10 module ppokFDM_m_write 36 3.1.11 module ppokFDM_m_write 77 3.2 stein3d1s 30 3.1 module pokFDM_m_write 77 3.2 stein3d1s 30	3.1.7 module ppohFDM_m_report	25.
1.9 model: pp04FDM_m_stab. 30. 1.10 model: pp04FDM_m_stab. 30. 1.11 model: pp04FDM_m_stab. 37. 3.11 model: pp04FDM_m_stab. 37. 2.1 model: pp04FDM_m_stab. 39. 2.1 model: pp04FDM_stab. 50.	3.1.8 module ppohFDM_m_source	27.
3.1.10 module ppohFDM_m_warfbe. 38 3.1.11 module ppohFDM_m_warfbe. 37 3.2 steina 341 39 3.2 steina 341 39 3.2 1 module ppohFDM_boundary. 50	3.1.9 module ppohFDM_m_stdlib	30-
3.1.11 module ppohFDM_m_swatch	3.1.10 module ppohFDM_m_surfbc	36-
3.2 seism3d3n	3.1.11 module ppohFDM_m_swatch	37.
3.2.1 module ppohFDM boundary. 50-	3.2 seism3d3n	39-
	3.2.1 module ppohFDM boundary.	50+

3.2.1 module ppohFDM_boundary.

Description

This module applying a zero-stress boundary condition on free surface. Zero stress value is applied to stress components (S pz, p=x,y,z) and the results of spatial derivatives just above and below the free-surface boundary are recalculated by using an one-side differentiation scheme. This scheme can treat irregular boundary as well as a flat boundary.

Dependency.

use ppohFDM_stdio

use ppohFDM_param

subroutine ppohFDM_bc_zero_stress.

(KFSZ, NIFS, NJFS, IFSX, IFSY, IFSZ, JFSX, JFSY, JFSZ).

Description

Applying zero stress value to stress components (S pz, p=x,y,z) on free surface.-

Arguments

integer, intent(in) :: KFSZ(NXP0:NXP1,NYP0:NYP1) ! depth of the free surface integer, intent(in) :: NIFS, NJFS ! number of points in x and y directions to examine free surface conditions-

integer, intent(in) :: IFSX(NFSMAX), IFSY(NFSMAX), IFSZ(NFSMAX)

evaluation point of free surface condition in x,y,z

integer, intent(in) :: JFSX(NFSMAX), JFSY(NFSMAX), JFSZ(NFSMAX)-

- ・ MITライセンス
 - 公開されているコードはユーザが自由に手を加えることができる

ppOpen-APPL/FDM 計算手順



1. 入力パラメータの例 (計算パラメータ)

- 1. 計算パラメータ (m_param.f90)
- モデルサイズ: 128*128*128 格子サイズ: 0.5 時間間隔: 0.025 MPI領域分割: 2*2*2

!-- << Model Size and Grid Width >> = 128 integer, parameter :: NX integer, parameter :: NY = 128 モデルサイズ integer, parameter :: NZ = 128 integer, parameter :: KFS = 25 integer, parameter :: NX1 = NX+1integer, parameter :: NY1 = NY+1integer, parameter :: NZ1 = NZ+1 integer, parameter :: NTMAX = 2000 integer, parameter :: NWRITE = 10 real(PN), parameter :: DX = 0.5 PN 格子間隔 real(PN), parameter :: DY = 0.5 PN real(PN), parameter :: DZ = 0.5 PN 時間間隔 = 0.025 PN real(PN), parameter :: DT integer, parameter :: NDUMP = 5

= 2

= 2

= 2

!-<< Parallel >>
integer, parameter :: IP
integer, parameter :: JP

integer, parameter :: KP integer, parameter :: NP integer, parameter :: NL MPI領域分割

= IP*JP*KP ! Number of process= 4 ! Order of the fd scheme





	muumumuai
4	# 地下構造の層の数
20 2.3 3.0 1.7	# 深さ(km) <i>,</i> 密度 (t/m3),
	P波速度(km/s) <i>,</i> S波速度 (km/s)
30 2.3 3.3 2.3	
40 2.7 5.0 3.3	
50 2.7 6.0 4.0	

modium dat

1. 入力パラメータの例 (観測点の設定)





(注意)./examples/seismic_3D-exampleにパラメータファイルが用 意されています。 ./src/seismic_3D/1.pureMPI-ppohVIS or 2.pure-MPIにCOPY

14

最大周波数と安定条件

(1)最大周波数の決め方



$$V_s^{min} = fm^{ax} imes \lambda^{min}$$

(伝わる速度) (周波数) (波長)

$$1.7 = fm^{ax} \times 3.0$$

$$f^{max} = \mathbf{0.6}(Hz)$$

(2) Δtの決め方

$$\Delta t < 0.2 \frac{\Delta x}{V_{max}}$$
 $\Delta t < 0.2 \frac{0.5}{4.0} = 0.025$

青: 未知数 黒: 既知数

2. 計算手順



3. 出力および可視化

出力データ

- 各MPIランクで出力されている

SEISM3D3. prm SEISM3D3. SPS. 000. 000. 000 SEISM3D3. SPS. 000. 000. 001 SEISM3D3. SPS. 000. 001. 000 SEISM3D3. SPS. 000. 001. 001 SEISM3D3. SPS. 001. 000. 000 SEISM3D3. SPS. 001. 000. 001 SEISM3D3. SPS. 001. 001. 000 SEISM3D3. SPS. 001. 001. 001 SEISM3D3. SPS. 001. 001. 001 SEISM3D3. SVR. 000. 000. 000 SEISM3D3. SUR. 000. 000. 001 SEISM3D3. SUR. 000. 001. 000 SEISM3D3. SUR. 000. 001. 001 SEISM3D3. SUR. 001. 000. 000 SEISM3D3. SUR. 001. 000. 001 SEISM3D3. SUR. 001. 001. 000 SEISM3D3. SUR. 001. 001. 001 SEISM3D3. WAV. 000. 000. 000 SEISM3D3. WAV. 000. 000. 000 SEISM3D3. XY. 000. 000. 000 SEISM3D3. XY. 000. 001. 000 SEISM3D3. XY. 001. 000. 000 SEISM3D3. XY. 001. 001. 000 SEISM3D3. XZ. 000. 000. 000 SEISM3D3. XZ. 000. 000. 001 SEISM3D3. XZ. 001. 000. 000 SEISM3D3. XZ. 001. 000. 001 SEISM3D3. YZ. 000. 000. 001 SEISM3D3. YZ. 000. 001. 000 SEISM3D3. YZ. 000. 001. 001

SEISM3D3.prm SEISM3D3.WAV.*** SEISM3D3.SPS.*** SEISM3D3.SUR.*** SEISM3D3.XY(XZ, YZ).***

計算パラメータ

観測点における波形

P波とS波の波動場

表面上での波動場

各断面での波動場

3. 出力および可視化 (cont.)

- ./tools/seismic_3D-toolsにおいてmakeすると4つ実行ファイル(catsnap, catwav, ppmxy3d3,rwav3d)が生成される
- % catsnap SEISM3D3.prm → 分割されたファイルが結合される(波動場)
- % catwav SEISM3D3.prm → 分割されたファイル が結合される(波形)
- % ppmxy3d3 → 波動場のスナップショット (xvやimagemagickで可視化)
- % rwav3d → 観測点の波形 (gnuplotで可視化)



可視化 ppOpen-MATH/VIS

- プロジェクト内で公開している大規模データのための可視化ライ ブラリppOpen-MATH/VIS
- ppOpen-APPL/FDM ver0.2.0に実装済み

 - 出力ファイルは ./src/seismic_3D/1.pureMPI-ppohVIS
 - control.datのMaxVoxelCountやMaxRefineLevelの値を大きくすると細かくなる(./examples/seismic_3D-example)
 - AVSやParaviewで可視化することができる



VXの速度場

ppOpen-APPL/FDM Performance in FX10

Weak scaling test in flat MPI

- モデルサイズ: 64^3 , 128^3 grid points
- 並列数:16~1024 コア
- 3D domain partitioning of MPI



Strong scaling test in the flat MPI



- 計算時間はコア数が増加することで低減している
 - 256³ grid points
 - 1024コアでの計算時間は、16コアの計算時間よりも18倍速くなっていた
 - 1024³ grid points
 - 1024コアでの計算時間は、128コアの計算時間よりも7倍速くなっていた。

Storing scaling test in the MPI/OpenMP hybrid parallel computing on FX10

- パフォーマンステストは8ノード(128コア,16コア/ノード)を使って評価
 P128T1, P64T2, P32T4, P16T8, P8T16 (Pはプロセス、Tはスレッド)
- モデルサイズ

- 256^3, 512^3グリッドポイント

すべてのモデルにおいて、最小の計算時間は、P64T2による並列のときであった
 Dure MPI Fult PGATE の古が2位直流化していた





ppOpen-APPL/FDMの実例 大規模連成計算 (地震動)

大規模連成計算の取り組み

ppOpen-HPCプロジェクト内で開発されているppOpen-MATH/MP couplerを用いて差分法(地震動→ ppOpen-APPL/FDM)と有限要素法(構造解析→ppOpen-APPL/FEM)を連成し、大規模連成計算をおこなっている



FDM: Seismic Wave Propagation ppOpen-MATH/MP

FEM: Building Response

地震動計算



淡路を震源とした地震を想定し、ポートアイランドにある計算科学研究機構 京コンピュータの建屋への影響をシミュレーションをおこなう

計算領域(黒点線):100km×100km×30km 計算格子間隔:Δx = Δy = 40 m, Δz = 20 m 計算ステップ: 36万ステップ(Δt=0.001, 360s) 計算グリッド: 2500×2500×1500

計算メモリ: 1.2TB



ppOpen-APPL/FDM 利用方法と実習

はじめに

- ppOpen-HPC project (URL: http://ppopenhpc.cc.utokyo.ac.jp/wordpress/) からppOpen-APPL/FDM ver0.2.0をダウンロード
- Paraviewをダウンロード、インストール

FX10の利用支援ポータルサイトからFUJITSU
 Software Development Tools をダウンロード、イン
 ストール

ファイルの解凍

 ダウンロードしたら解凍する % tar zxvf ppohFDM_0.2.0.tar.gz

[c31003@oakleaf-fx-4 ppohFDM_0.2.0]\$ ls								
doc	INSTALL_ppohAPPL-FDM	Makefile	ppohMATH-VIS-install	src				
etc	LICENSE_ppohAPPL-FDM	Makefile.in	ppohMATH-VIS-lib	tools				
Examples	LICENSE_ppohMATH-VIS	README_pp	oh-APPL-FDM					

- 以下このディレクトリをルートディレクトリ(/)として説明 する
- サンプルのパラメータファイルは、以下においてあり ます

./src/example/seismic_3D-example

入力パラメータ設定(1)

- 入力パラメータファイル(3つ)
 - 1. 観測点(X,Y,Z(km))の設定



2. ソースパラメータを設定



30

入力パラメータ設定(2)

- 地下構造モデル(深さ(depth),密度(RO),P波の速度 (VP),S波の速度(VS))を設定
 - medium.dat

medium.dat



計算パラメータ設定

integer, parameter :: KP

計算パラメータ (m_param.f90)

(計算モデルとタイムステップ)

- モデルサイズ:NX, NY, NZ
- 格子間隔: DX, DY, DZ
- ・ タイムステップ:NTMAX
- 時間間隔:DT

(MPI: 3次元分割)

- 分割: IP, JP, KP
- プロセス数: NP

!-- << Model Size and Grid Width >> = 128 integer, parameter :: NX integer, parameter :: NY = 128 モデルサイズ = 128 integer, parameter :: NZ = 25 integer, parameter :: KFS integer, parameter :: NX1 = NX+1integer, parameter :: NY1 = NY+1integer, parameter :: NZ1 = NZ+1 integer, parameter :: NTMAX = 2000 タイムステップ integer, parameter :: NWRITE = 10 real(PN), parameter :: DX = 0.5 PN 格子間隔 real(PN), parameter :: DY = 0.5 PN real(PN), parameter :: DZ = 0.5 PN 時間間隔 = 0.025 PN real(PN), parameter :: DT integer, parameter :: NDUMP = 5 !-<< Parallel >> integer, parameter :: IP = 2 MPI領域分割 integer, parameter :: JP = 2

= 2

integer, parameter :: NP = IP^*JP^*KP ! Number of process

(注意)./examples/seismic_3D-exampleにパラメータファイルが用意されています。 ./src/seimic_3D/2.pureMPIにコピーしてください

コンパイル pureMPIとハイブリッド並列

- ・コンパイル
 - ./src/seism_3D/2.pureMPIや./src/seism_3D/3.hybrid にあるMakefile.optionをIntel Compiler 用→ FX10用に コメントアウトする
 - % ./make seism3d-mpi, %./make seism3d-hybrid
 - ・実行ファイルseism3d3nが生成されていることを確認



- ・バッチ実行
 - % pjsub job
 - % pjstat
 - ・実行されているか確認



job file in the hybrid parallel

#!/bin/sh

```
#PJM -L "rscgrp=short"
#PJM -L "node=4"
#PJM -L "elapse=00:30:00"
#PJM -g **
#PJM -g **
```

```
export OMP_NUM_THREADS=16
```

mpiexec ./seism3d3n

コンパイルと実行 with ppOpen-MATH/VIS

- ・コインパイル
 - 1. % ./make
 - 2. % ./make install
 - 3. % ./make seism3d-ppohVIS
 - 4. % ./make install

./src/seismic_3D/1.pureMPI-ppohVISに実行ファイルseism3d3nが 生成されていることを確認

- Makefile.inにMATH/VISライブラリのコンパイル先が書かれている。各ユーザはディレクトリを場所を絶対パスで書く
 デフォルトは /usr/local/ppoh-HPCになっている
- ジョブ投入は、puerMPIと同様

演習(1)

- 1. 媒質の物理値を変化させ、波動場と波形の観測せよ。パラメータは以下の通り
 - 計算パラメータの設定(m_param.f90)
 - モデルサイズ(NX*NY*NZ):128*128*128
 - 媒質の設定(medium.dat)



- ソースの設定(source.dat)



演習(1)(cont.)

- 観測点の設定(station.dat)

4 10.0 10.0 0.0 40.0 10.0 0.0 10.0 40.0 0.0 40.0 40.0 0.0

- 上記のパラメータとデフォルトパラメータとの違い
 を確認せよ
 - 1. 媒質のパラメータを変更したときの違い
 - 2. ソースのパラメータを変更したときの違い
 - 3. 両方(媒質+ソース)のパラメータを変更したときの違い

演習(1)(cont.)

- ・コンパイル
 - % ./make seism3d-mpi
- ・実行
 - % cd ./src/seimic_3D/2.pureMPI/
 - %pjsub job

#!/bin/sh

```
#PJM -L "rscgrp=**"
#PJM -L "node=**"
#PJM -L "elapse=**:**:**"
#PJM -g **
#PJM -g **
```

mpiexec lpgparm -p 256MB -d 256MB -h 256MB -s 256MB -t 256MB ./seism3d3n

演習(1)(cont.)

- ローカルPCで可視化
 - ./tools/seismic_3D-toolsにおいてmakeすると4つ実行ファイル (catsnap, catwav, ppmxy3d3,rwav3d)が生成される
- 波動場の確認
 - % catsnap SEISM3D3.prm
 - % ppmxy3d3 –f SEISM3D3.prm –tim –tick –pall –pmul 1e3 –smul 1e3 –ptype SPS
 - % xv *.ppm
- 波形の確認
 - % catwav SEISM3D3.prm
 - % rwav3d SEISM3D3.WAV > wave.dat
 - % gnuplot
 - plot "wave.dat" index0 using 4:5 with lines lw 2

演習(1a)

- 1. 媒質の物理値を変化させ、波動場の観測せよ。パ ラメータは以下の通り
 - 計算パラメータの設定(m_param.f90)
 - モデルサイズ(NX*NY*NZ):128*128*128
 - 媒質の設定(medium.dat)



- ソースの設定(source.dat)



演習(1a)(cont.)

- 観測点の設定(station.dat)

4 10.0 10.0 0.0 40.0 10.0 0.0 10.0 40.0 0.0 40.0 40.0 0.0

control.dat

[Refine] AvailableMemory = 32.0 MaxVoxelCount = 5000 MaxRefineLevel = 1000

- 可視化したい物理値はseism3d3n.f90のL107において設定する
- 上記のパラメータとデフォルトパラメータとの違いを確認せよ
 - 1. 媒質のパラメータを変更したときの違い
 - 2. ソースのパラメータを変更したときの違い
 - 3. control.datのMaxVoxelCountとMaxRefineLevelの値を変更したときの違い確認せよ

演習(1a)(cont.)

・ コンパイル

- 1. % ./make
- 2. % ./make install
- 3. % ./make seism3d-ppohVIS
- 4. % ./make install

• 実行

- % cd ./src/seimic_3D/1.pureMPI-ppohVIS
- %pjsub job

#!/bin/sh

```
#PJM -L "rscgrp=**"
#PJM -L "node=**"
#PJM -L "elapse=**:**:**"
#PJM -g **
#PJM -g **
```

mpiexec lpgparm -p 256MB -d 256MB -h 256MB -s 256MB -t 256MB ./seism3d3n

演習(1a)

- ppOpen-MATH/VISで出力されたinpファイルを paraviewを使って可視化
 - ./src/seismic_3D/1.pureMPI-ppohVIS/ppohVISにデータ が出力されていることを確認
- Paraviewを使って可視化する

演習(2) pureMPI

- 並列数とモデルサイズを変更して計算時間の変 化を計測せよ
 - m_param.f90
 - モデルサイズ: 256*256*256 grid points
 - NP: 8, 16, 32, 64, • •
 - プロファイル情報を取得するために、計測した部分に call fapp_start, call fapp_stopで挟む
 - seism3d3n.f90に書き込む(デフォルトでコメントアウトされている)

!! Velocity Update
call fapp_start("region2",1,1)
call ppohFDM_update_vel (1, NXP, 1, NYP, 1, NZP)
call fapp_stop("region2",1,1)

演習(2) pureMPI

・ Job ファイル

#!/bin/sh

#PJM -L "rscgrp=**"
#PJM -L "node=**"
#PJM -L "elapse=**:**:**"
#PJM -g **
#PJM -g **

fapp -C -d prof -L 1 -Ihwm -Hevent=Statistics mpiexec lpgparm -p 256MB -d 256MB -h 256MB -s 256MB -t 256MB ./seism3d3n

プロファイルを保存するディレクトリが必要 _____/src/seismic_3D/2.pureMPI/prof

(注意) 計測にあたりseism3d3n.f90のL316にあるcall ppohFDM_io_write() をコメントアウトする

演習(3) Hybrid parallel computing

- 1. 並列数とモデルサイズを変更して計算時間の変化を計測 せよ
 - m_param.f90
 - モデルサイズ: 256*256*256 grid points
 - 使用するノード数: 8ノード(128コア) or 4ノード(64コア)に固定
 - IP, JP, KPの値を変化させる(プロセス数)
 - Job文のexport OMP_NUM_THREADSの値を変化させる
 - Jobファイル

**はユーザがパラメータに応じて変更

#PJM -L "rscgrp=***"
#PJM -L "node=**"
#PJM -L "elapse=**:**:**"
#PJM -g **
#DIM ______**"

#!/bin/sh

```
#PJM --mpi "proc=**"
```

```
export OMP_NUM_THREADS=**
```

fapp -C -d prof -L 1 -Ihwm -Hevent=Statistics mpiexec lpgparm -p 256MB -d 256MB -h 256MB -s 256MB -t 256MB ./seism3d3n

参照文献

- 古村孝志, 地震波伝播と強震動の大規模並列
 FDMシミュレーション, 東京大学情報基盤センター
 スーパーコンピューティングニュース, Vol11,
 pp.35-63, 2009.
- ppOpen-APPL/FDM ver0.2.0 user guide
- ppOpen-APPL/FDM ver0.2.0 reference guide