

Ani3D расширение параллельной платформы INMOST для CFD приложений

В. К. Крамаренко^{1,3} И. Н. Коньшин^{1,2} Ю. В. Василевский^{1,3}

¹ИВМ РАН

²ФИЦ ИУ РАН (ВЦ РАН)

³МФТИ

CFD-Weekend, 2017

План

- Последовательный пакет Anis3D
- Параллельная платформа INMOST
- Интеграция Anis3D–INMOST
- Модельные CFD задачи

Advanced Numerical Instruments 3D

Secure | <https://sourceforge.net/projects/ani3d/>

sourceforge Search [Browse](#) [Enterprise](#) [Blog](#) [Articles](#) [Deals](#) [Help](#)

SOLUTION CENTERS Resources Newsletters Cloud Storage Providers Business VoIP Providers Call Center Providers

[Home](#) / [Browse](#) / [Development](#) / [Software Development](#) / [Advanced Numerical Instruments 3D](#)



Advanced Numerical Instruments 3D

Advanced numerical instruments: adaptive meshing, FE methods, solvers

Brought to you by: [lipnikov](#), [vasilevs](#)

[Summary](#) [Files](#) [Reviews](#) [Support](#) [Wiki](#) [Mailing Lists](#) [Bugs](#) [Discussion](#)

★ [Add a Review](#)

↓ [4 Downloads](#) (This Week)

📅 [Last Update: 2017-03-12](#)

sf

Download

ani3D-3.1.tar.gz

[Browse All Files](#)

[Tweet](#) [G+](#)



Description

Ani3D software package

- Generation of tetrahedral meshes
- Mesh adaptation
- FEM discretization on tetrahedral meshes
- Solution of linear and nonlinear systems
- *Serial* code in Fortran and C

Open source code: <http://sourceforge.net/projects/ani3d>

Ani3D libraries

Ani-MBA library

- generation of quasi-uniform meshes in a user-defined metric
- uniform refinement of tetrahedral meshes

Ani-FEM library

- local finite element discretization on tetrahedron
- assembling the local discretizations into a global linear system

the rest of libraries...

- Ani-C2F, Ani-INB, Ani-LMR, Ani-PRJ, Ani-RCB
- Ani-ILU, Ani-LU

INMOST v0.1 - igor.konshi... INMOST hpc oilgas - Поиск в Goo... Суперкомпьютерные техно... Скорость интернет соедин...

www.inmost.org

View on GitHub

INMOST

A toolkit for distributed mathematical modeling

tar.gz .zip

Welcome to INMOST project page.

You are advised to have the following directory structure of INMOST resources for further instructions to work:

```
INMOST.pages
INMOST.wiki
INMOST.lib
```

INMOST.pages will be used for the site, INMOST.wiki will be used for Wiki resource and INMOST.lib for the library itself.

Setup the repository of this site on your computer

The screenshot shows a web browser window displaying the GitHub repository page for `INMOST-DEV/INMOST`. The browser's address bar shows the URL `https://github.com/INMOST-DEV/INMOST/wiki`. The GitHub header includes the repository name, a search bar, and navigation links for `Explore`, `Features`, `Enterprise`, and `Blog`. There are `Sign up` and `Sign in` buttons. Below the header, the repository name `INMOST-DEV / INMOST` is displayed, along with `Watch 2`, `Star 0`, and `Fork 0` statistics. The main content area features a `Home` section with the text `Kirill Terekhov edited this page 4 days ago · 12 revisions`. A large heading `Welcome to the INMOST wiki!` is followed by a list of links: `Compiling INMOST:`, `Compilation guides`, `Reporting issues and preparing tests:`, `Guide for testing`, `Explore included examples:`, `List of Examples`, and `Please read before writing Wiki articles:`. On the right side, there is a `Pages 29` dropdown menu with a search bar `Find a Page...` and a list of page titles: `0100 Compilation`, `0200 Compilation Windows`, `0201 Obtain MSVC`, `0202 Obtain MSMPI`, `0203 Compilation INMOST Windows`, `0204 Compilation Parmetis Windows`, `0205 Compilation Zoltan Windows`, and `0206 Compilation PFTSc`.

The screenshot shows a web browser window with the URL `www.inmost.org/Doxygen/html/annotated.html`. The page title is "INMOST" and the subtitle is "A toolkit for distributed mathematical modeling". The navigation menu includes "Main Page", "Related Pages", "Classes", and "Files". The "Classes" menu is expanded, showing "Class List", "Class Index", "Class Hierarchy", and "Class Members". The "Class List" page displays a list of classes and interfaces with brief descriptions. The classes listed are:

- INMOST** (Namespace)
- Automatizator** (Class)
 - table
 - expr
 - TagMemory
 - Tag
- TagManager** (Class)
 - sparse_sub_record
- Storage** (Class) - Base class for `Mesh`, `Element`, and `ElementSet` classes
- reference_array** (Class) - Storage type for representing arrays of `Element` references
 - const_iterator
 - const_reverse_iterator
 - iterator

[detail level 1 2 3 4]

INMOST

- **I**ntegrated
- **N**umerical
- **M**odelling and
- **O**bject-oriented
- **S**upercomputing
- **T**echnologies

INMOST is the software platform for developing parallel numerical models on general meshes.

INMOST is a tool for supercomputer simulations characterized by a maximum generality of supported computational meshes, distributed data structure flexibility, cost-effectiveness, cross platform portability.

INMOST: Mesh

- Mesh data are distributed
- Conformal meshes with Tetrahedra, Hexahedra, Prisms, Pyramids, Polyhedra etc.
- Mesh elements hierarchy: Vertex, Edge, Face, Cell
- Mesh element can contain some data (tags)
- Specification of “ghost” elements (“hallo”)
- Exchange tag data for ghost elements
- Save/Load mesh data in a parallel format file (.pmf, .pvtk, ...)

INMOST: Solver

- Assemble the distributed matrix and right-hand side of the linear system
- Parallel solution of the distributed linear system
- A set of internal linear solvers
- A set of external solvers: PETSc, Trilinos, SuperLU, ...

Parallelization Technology

Mesh

- read and partition
- refine on each processor *preserving conformity*
- merge

Init

- enumerate respective elements
- create tags for DOFs numbers
- synchronize tags

Assemble

- generate local matrix for each tetrahedron
- assemble them into a global matrix

Stokes problem

Formulation

$$\begin{aligned} -\Delta \mathbf{u} + \nabla p &= 0 & \text{in } \Omega \\ \nabla \cdot \mathbf{u} &= 0 & \text{in } \Omega \\ \mathbf{u} &= \mathbf{u}_0 & \text{on } \partial\Omega_1 \\ \mathbf{u} &= 0 & \text{on } \partial\Omega_2 \\ \frac{\partial \mathbf{u}}{\partial \mathbf{n}} - p &= 0 & \text{on } \partial\Omega_3 \end{aligned}$$

$$\mathbf{u}_0 = (64 \cdot (y - 0.5) \cdot (1 - y) \cdot z \cdot (1 - z), 0, 0)$$

Stokes problem

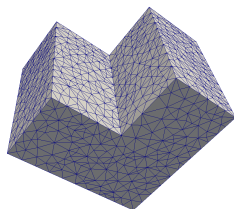


Figure : The coarsest mesh S0

Table : The problems parameters

Problem name	S0	S1	S2
Number of nodes	5187	36824	279903
Number of edges	31637	243079	1908542
Number of tetrahedra	25113	200904	1607232
Matrix size	115659	876533	6845238
Number of nonzeros	10751851	84374191	668849086

Numerical results

Table : The solution of S1 problem on $p = 1, \dots, 32$ processors

p	T_{ini}	T_{ass}	T_{prec}	T_{iter}	N_{iter}	Dens	PM	T_{sol}	S
1	0.38	69.07	31.41	97.20	242	0.85	0	128.61	1.00
2	0.28	40.97	20.30	76.88	322	0.93	0	97.18	1.32
4	0.21	25.31	12.76	47.88	332	1.03	0	60.64	2.12
8	0.14	13.76	7.83	28.68	332	1.15	1	36.51	3.52
16	0.10	8.37	4.77	18.46	362	1.37	3	23.23	5.53
32	0.06	5.09	4.12	12.70	402	1.65	8	16.82	7.64

Table : The solution of S2 problem on $p = 4, \dots, 32$ processors

p	T_{ini}	T_{ass}	T_{prec}	T_{iter}	N_{iter}	Dens	PM	T_{sol}	S
4	1.48	181.05	142.29	1484.02	722	0.97	0	1626.31	1.00
8	0.90	94.03	76.01	847.62	802	1.03	0	923.63	1.76
16	0.58	52.03	72.70	481.50	802	1.10	2	554.20	2.93
32	0.37	29.23	26.88	288.73	802	1.21	2	315.61	5.15

Unsteady convection-diffusion problem

Formulation

$$\begin{aligned} \frac{\partial c}{\partial t} - \nabla(D\nabla c) + \mathbf{v} \cdot \nabla c &= 0 \quad \text{in } \Omega \\ c &= g \quad \text{on } \partial\Omega \end{aligned}$$
$$D = \begin{pmatrix} 0.001 & 0 & 0 \\ 0 & 0.001 & 0 \\ 0 & 0 & 0.001 \end{pmatrix}$$
$$\mathbf{v} = (1, 0, 0)$$

Unsteady convection-diffusion problem

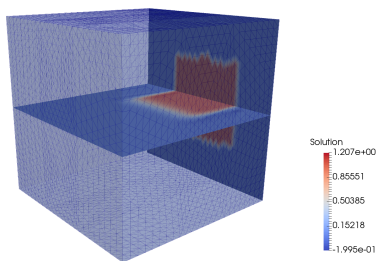


Figure : The concentration at time $t = 0.5$

Table : The problems parameters

Problem name	L0	L1	L2
Number of nodes	20417	155905	1218561
Number of tetrahedra	111616	892928	7143424
Matrix size	20417	155905	1218561
Number of nonzeros	291393	2281217	18053121

Numerical results

Table : The solution of problem on mesh L1 on $p = 1, \dots, 32$ processors

p	T_{ini}	T_{ass}	T_{sol}	T_{Σ}	S
1	26.16	552.87	33.86	612.89	1.00
2	16.48	327.93	21.31	365.72	1.67
4	9.83	197.89	12.19	219.91	2.78
8	5.75	114.81	7.28	127.84	4.79
16	3.56	73.57	4.03	81.16	7.55
32	2.31	47.28	2.45	52.04	11.77

Table : The solution of problem on mesh L2 on $p = 1, \dots, 32$ processors

p	T_{ini}	T_{ass}	T_{sol}	T_{Σ}	S
1	436.82	5692.31	723.23	6852.36	1.00
2	169.17	2628.09	258.71	3055.97	2.24
4	94.33	1461.24	147.30	1702.87	4.02
8	53.62	874.21	91.24	1019.07	6.72
16	31.52	522.99	53.28	607.79	11.27
32	17.6	308.84	29.61	356.05	19.24

Выводы

- Представлено Ani3D-расширение параллельной программной платформы INMOST.
- Функциональность INMOST расширена за счет КЭ и сеточной библиотек пакета Ani3D.
- Численные эксперименты продемонстрировали параллельную эффективность подхода.
- Примеры могут быть загружены с:

https://github.com/INMOST-DEV/INMOST/tree/master/Examples/Ani_Inmost

Динамическое управление параметрами линейных решателей для нестационарной задачи фильтрации

Д. В. Багаев^{1,3} И. Н. Коньшин^{1,2} К. Д. Никитин¹

¹ИВМ РАН

²ФИЦ ИУ РАН (ВЦ РАН)

³МГУ

CFD-Weekend, 2017

Linear systems and linear solvers

- Solution of linear systems
- Linear systems are differ
- A lot of linear solvers
- A lot of parameters for each linear solver
- How to choose the linear solver
- How to choose the linear solver parameters

Goal

- Time optimization of $A_k x = b_k$
- INMOST
- BILU2 linear solver
- τ – threshold parameter p
- q – overlap parameter

N14 sample problem

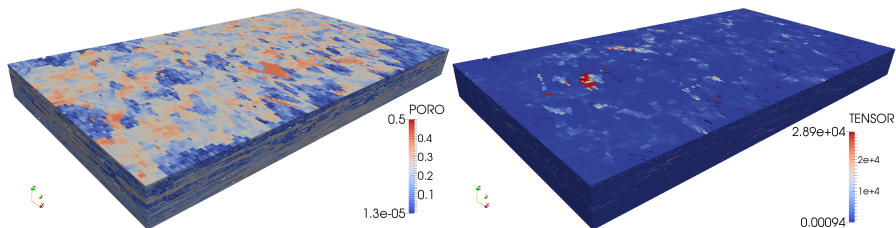


Figure : The porosity and permeability distributions for SPE-10 problem

Black-Oil Simulator for Scholars (BOSS) for SPE-10 problem.
The size of the model mesh is $60 \times 220 \times 85$ cells ($1.122 \cdot 10^6$ cells).
The porosity varies from $1.3 \cdot 10^{-5}$ to 0.5 (see Fig. left).
The permeability varies from 10^{-3} to $3 \cdot 10^4$ (see Fig. right).
The model has 5 vertical wells completed throughout formation.
The dimension of the linear system N14 is 3 896 013 unknowns.

Unsteady black-oil simulation – fixed parameters (τ, q)

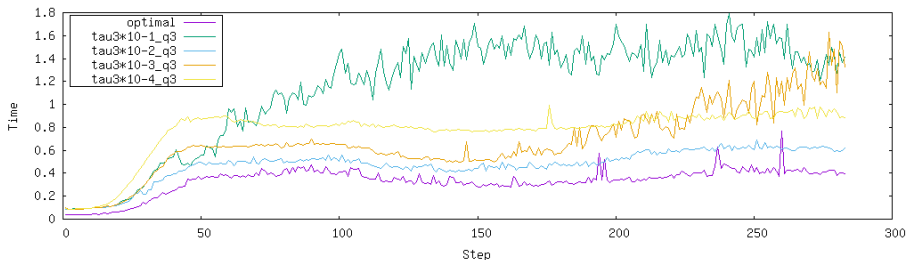


Figure : Unsteady black-oil simulation solution times depending on time step k

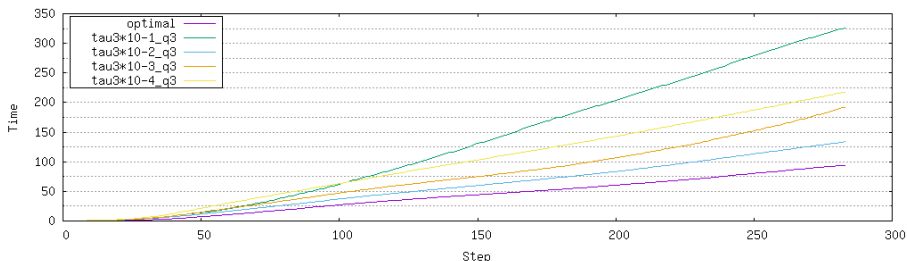


Figure : Unsteady black-oil simulation cumulative times depending on time step k

Unsteady black-oil simulation – parameters optimization

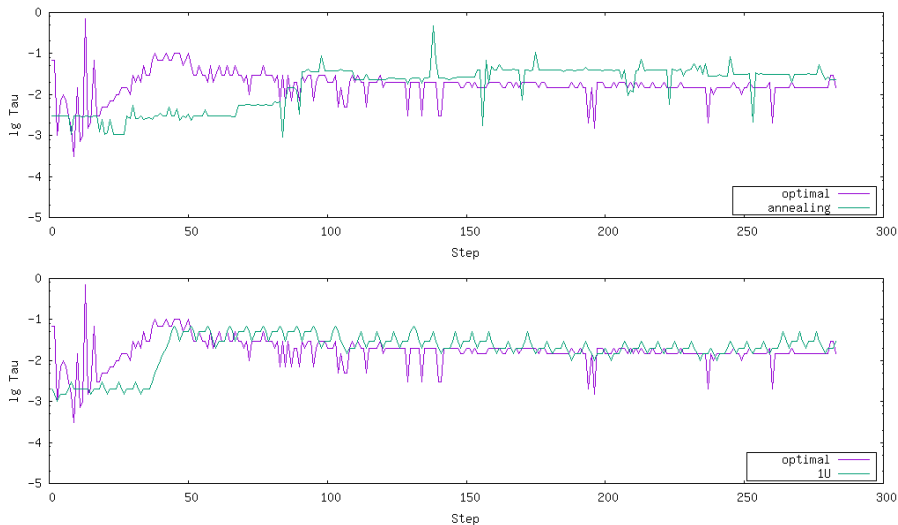


Figure : Optimizing τ for black-oil simulator

Unsteady black-oil simulation – parameters optimization

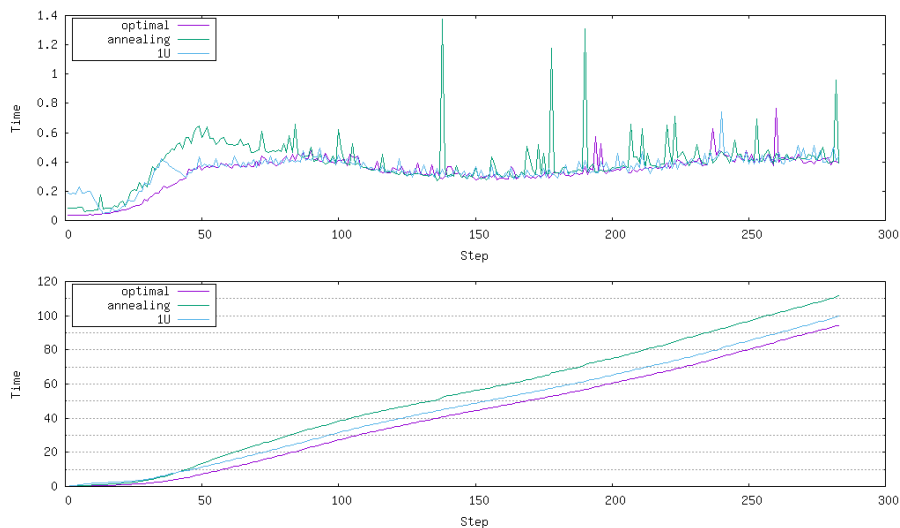


Figure : Local and cumulative times depending on time step k

Unsteady black-oil simulation – parameters optimization

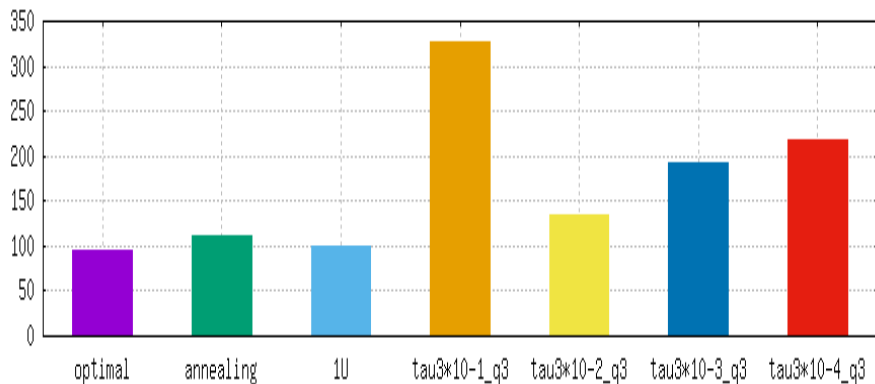


Figure : Cumulative times bar chart for default sets of parameters and for proposed algorithms compared with the optimal one

Выводы

- Предложены несколько алгоритмов оптимизации порога фильтрации τ
- Время решения близко ($\approx 10\%$) к оптимальному
- Лучше чем любой фиксированный набор параметров
 - ▶ В 2-3-4 раз лучше чем “обычные” наборы параметров
 - ▶ В 1.5 раза лучше чем “наилучший” набор
- Может быть применено к:
 - ▶ другим линейным решателям типа PETSc $AS(q)+ILU(k)$
 - ▶ другим нестационарным приложениям, например, гемодинамике