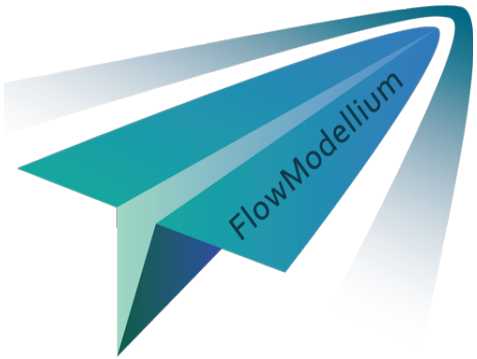
Parallel computational methods for high-speed aerodynamics on multi/many core Intel CPU

V.A. Titarev[1,2], A.V. Chikitkin[2], M.N. Petrov[2], S.V. Utyuzhnikov[2,3]

(1) FRC CSC RaS, (2) MIPT, (3) University of Manchester

IXPUG/RU Workshop, Saint-Petersburg, June 9-10

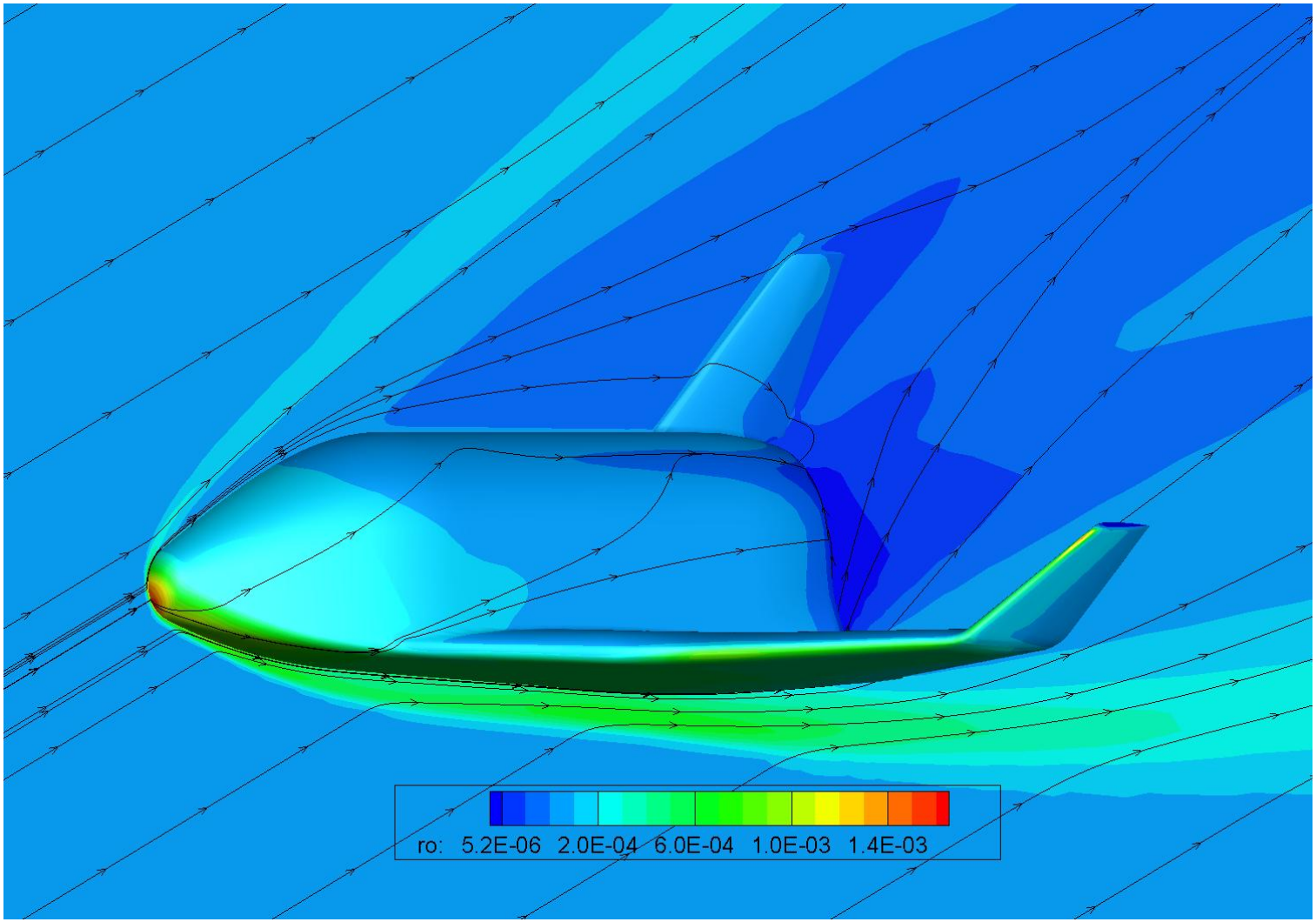


Introduction

1. Computational external aerodynamics of chemically reacting high-speed flows requires significant computing resources
2. Most of the existing codes are adapted for systems with 4-6 core CPUs
3. However, modern HPC systems use CPUs with significantly large number of cores as well as many-core Intel Xeon Phi coprocessors
4. Therefore, porting applied codes and implicit time-evolution methods to such systems is a very relevant problem
5. In our presentation we describe parallel implicit methods implemented in the FlowModellium code
6. We demonstrate scaling results on RSK Petastream systems.

Typical application: external flow

Flow conditions: $H = 70$ km, M_∞=25, angle of attack $\alpha = 25$ degrees.



Aerodynamics solver of the "FlowModellium" lab

1. The solver is intended for computational modelling of external high-speed flows in the large range of Re and Kn numbers
2. There is a state registration on V.A. Titarev and S.V. Utyuzhnikov, N. 2013619670
3. General applications:
   - Development of new numerical methods, including domain decomposition methods
   - Solving fundamental and applied problems
   - Development of parallel algorithms for multi & many-core systems.
4. Physical models available :
   - equilibrium chemistry
   - non-equilibrim chemistry (up to 11 components)
   - compressible Navier-Stokes equations, RANS models
5. Finite-volume methods implemented:
   - LSQ approximation on arbitrary meshes
   - Directional TVD & WENO schemes on hexahedrons
   - Convective fluxes up to 5th order
6. Time integration: explicit, implicit of LU-SGS type.

Governing equations

- We assume ideal-gas equation of state.
- In non-dimensional variables the equations take the following form

$$\frac{\partial}{\partial t}\boldsymbol{U} + \nabla(\boldsymbol{F} - \frac{1}{\mathrm{Re}_*}\boldsymbol{G}) = \boldsymbol{S}(\boldsymbol{U}), \quad \mathrm{Re}_* = \frac{\rho_* v_* L_*}{\mu_*}.$$

- Here $\boldsymbol{U}$ is the conserved variables vector, $\boldsymbol{F}$, $\boldsymbol{G}$ - tensors of convective and viscous fluxes, $\boldsymbol{S}$ - stiff source term due to chemistry.
- The exact form of the vectors is defined by the physical model.
- For example, for non-equilibrium chemistry the following components are used:

$$O, \quad N, \quad O_2, \quad N_2, \quad NO, \quad O^+, \quad N^+, \quad NO^+, \quad O_2^+, \quad N_2^+, \quad e.$$

- Then the length of vectors is 15 entries.

Time-marching implicit method for steady flows

- On an arbitrary mesh the implicit method has the following form:

$$\frac{\Delta \boldsymbol{U}_i}{\Delta t} = \boldsymbol{R}_i^{n+1}, \quad \Delta \boldsymbol{U}_i = \boldsymbol{U}_i^{n+1} - \boldsymbol{U}_i^n, \quad i = 1, 2, \ldots N_{cell}$$

$$\boldsymbol{R}_i^{n+1} = -\frac{1}{|V_i|} \sum_l \boldsymbol{\Phi}_{li}^{n+1} + \boldsymbol{S}(\boldsymbol{U}_i^{n+1}), \quad \boldsymbol{\Phi}_{li} = \int_{a_{li}} \boldsymbol{n} \cdot (\boldsymbol{F} - \frac{1}{\mathrm{Re}_*} \boldsymbol{G}) da.$$

- After linearization around low time level we get that time increment in a cell i is connected to its neighbors as

$$D_i \Delta \boldsymbol{U}_i + \frac{1}{2|V_i|} \sum_l \boldsymbol{\Psi}(\boldsymbol{U}_{\sigma_l(i)}, \Delta \boldsymbol{U}_{\sigma_l(i)}) = \boldsymbol{R}_i^n$$

- Serial LU-SGS algorithm on unstructured mesh is given by (Men'shov & Nakamura, 1996):
  - Backward sweep: for $i = N_{cell}, N_{cell} - 1, \ldots 1$

$$D_i \Delta \boldsymbol{U}_i^* = -\frac{1}{2|V_i|} \sum_{l:\sigma_l(i)<i} \boldsymbol{\Psi}(\boldsymbol{U}_{\sigma_l(i)}, \Delta \boldsymbol{U}_{\sigma_l(i)}^*) + \boldsymbol{R}_i^n$$

  - Forward sweep: for $i = 1, 2, \ldots N_{cell}$.

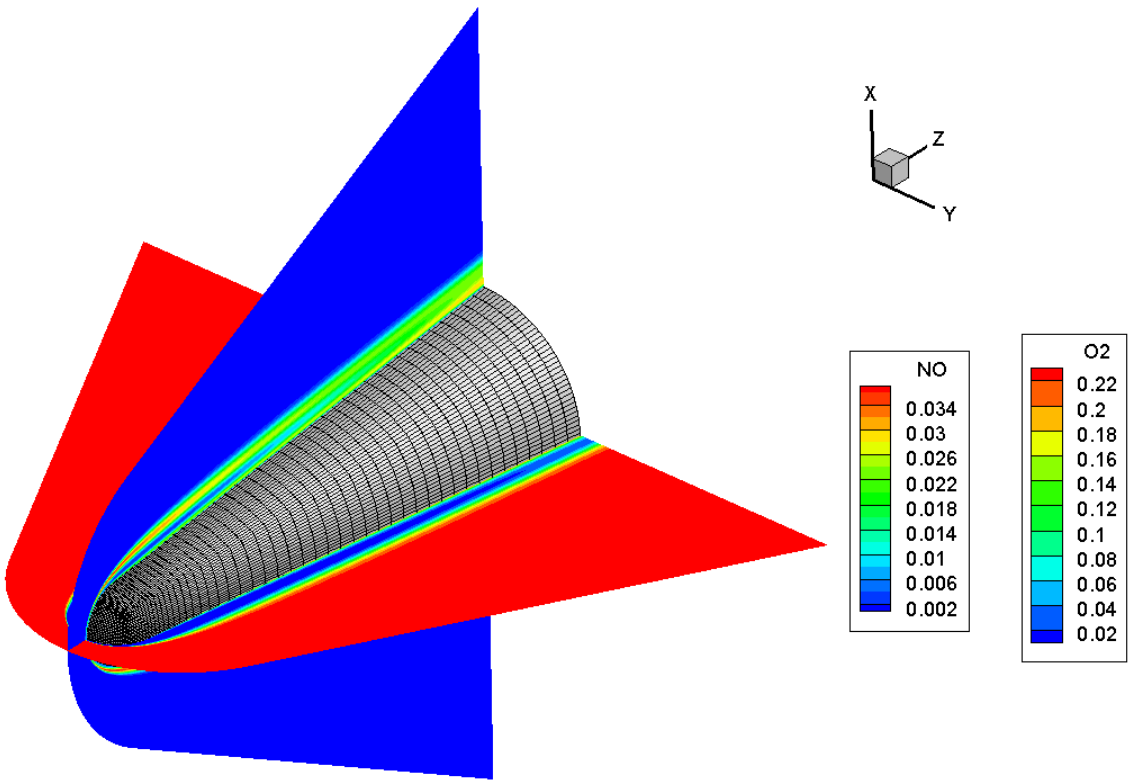
$$D_i \Delta \boldsymbol{U}_i = \Delta \boldsymbol{U}_i^* - \frac{1}{2|V_i|} \sum_{l:\sigma_l(i)>i} \boldsymbol{\Psi}(\boldsymbol{U}_{\sigma_l(i)}, \Delta \boldsymbol{U}_{\sigma_l(i)})$$

Two-level parallel model

1. Upper level - MPI between nodes/CPUs and geometrical decomposition of the spatial mesh:
   - external program (e.g. Metis or Kaporin's code) is used to split the mesh into non-overlapping blocks
   - our in-house pre-processor prepares multi-block mesh, storing local numbering of vertices, faces and cells
   - during computations ghost cell data is exchanged using MPI
2. Low level - OpenMP algorithm for shared-memory systems
3. We can use practically arbitrary combination of MPI processes/OpenMP threads
4. The main difficulty is to have a parallel version of LU-SGS which would be equivalent to its serial one. The serial LU-SGS method cannot be made parallel in a straightforward manner as forward and backward sweeps are loops with direct data dependencies.
5. We are testing an OpenMP version of LU-SGS, which is both equivalent to the serial one and scales well. Details are omitted.

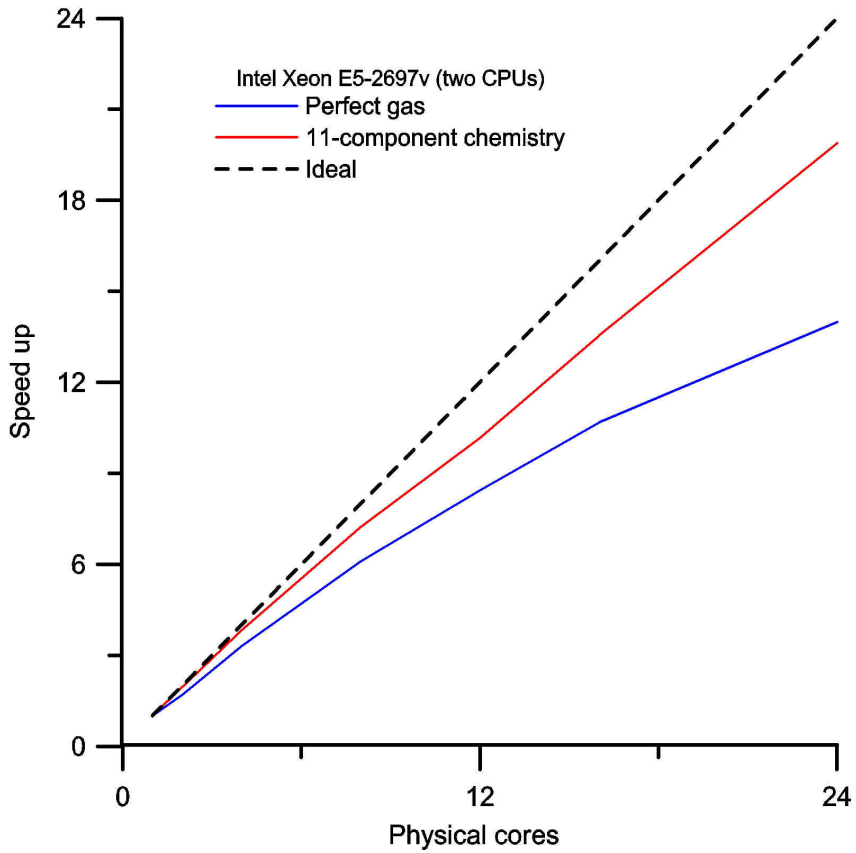
Numerical tests for the OpenMP implicit algorithm

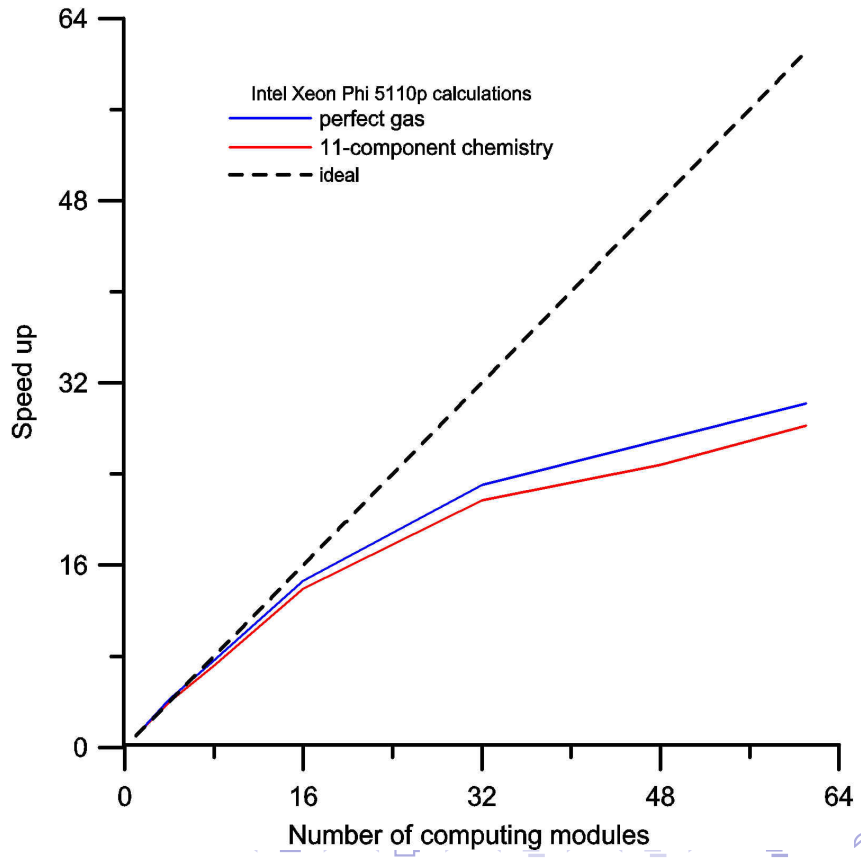
- Test problem: high-speed flow over model RAM-C geometry (blunted cone)
- Hexa mesh containes 600 000 elements. Calculations are run on
  - dual Intel Xeon E5-2697v sever (12 cores per CPU, 2.7 Ghz)
  - Intel Xeon Phi 5110p sever (61 computing module, 244 hyperthreads, 1.053 Ghz)
- Figure illustrates flow pattern for $M_\infty = 27$.



Numerical tests for the OpenMP implicit algorithm (2)

- Hardware: dual-CPU Intel Xeon E5-2697v server (left) and Intel Xeon Phi 5110p server (right)
- In our calculations Xeon Phi 5110p $\approx$ 10% faster single 12-core Xeon E5-2697v.
- Intel Xeon Phi 7110X $\approx$ equal dual 8-core Xeon E5-2690 server.



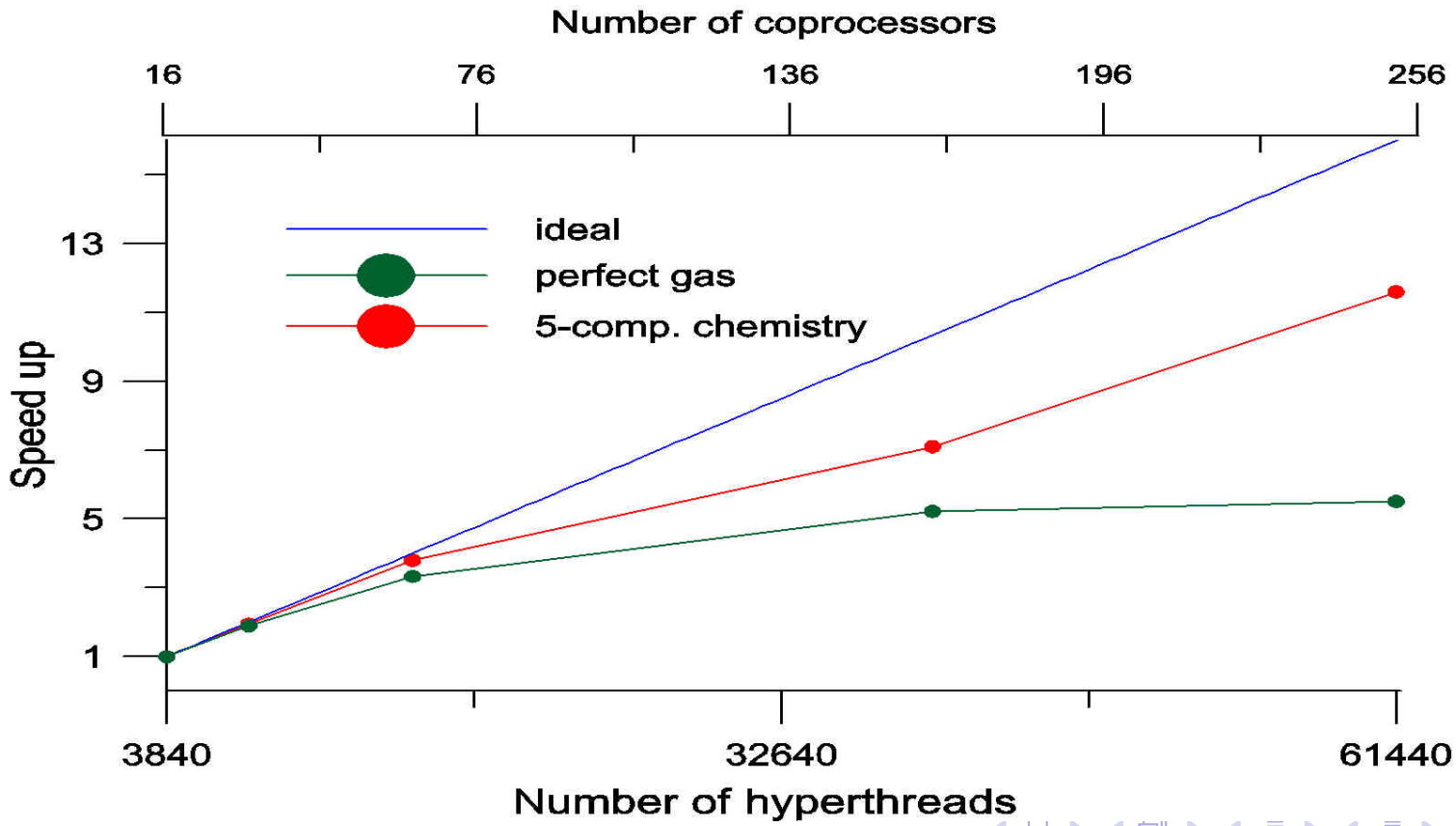


Numerical tests for industrial problem

- We use three-dimensional geometry of a spacecraft.
- Calculations are run for perfect gas model and 5-component single-temperature non-equilibrum chemistry
- Flow regimes $M_\infty = 10 \ldots 40$, angle of attack $\alpha = 25 \ldots 35$.
- Block-structured hexa mesh contained 5.5 million cells & 147 blocks, converted into unstructured hexa.
- One time step using chemistry takes
  - 2.3 sec on 16 Intel Xeon Phi 5110p
  - 1.8 sec on 12 Intel Xeon E5-2697v
- Computations with perfect gas model are about 5-10 times faster.
- It takes 3×10^5 time steps to converge to steady solution.

Calculations on RSK Petastream of SPbPU

- The system contains 256 Intel Xeon Phi 5120D (240 hyperthreads, 1.053 Ghz, 8 GB RAM per node), Intel Fortran 2016 & Intel MPI
- For perfect gas model speed up is 5.22 times from 16 to 128 coprocessors (65%).
- For chemistry speed up is 11.6 times from 16 to 256 coprocessors (74%).



Conclusions

Future plans:

- Further optimize MPI exchanges.
- Extended testing on tetrahedral and prismatic meshes
- Extended testing for kinetic solvers (preliminary results look encouraging).

The work was supported by

- Russian government under grant "On measures to Attract Leading Scientists to Russian Educational Institutions"(contract No. 11.G34.31.0072).
- RFFR project 15-07-02986
- Discussions with Intel Russia and RSK Group
- Computing resources of JSC and SPbPU

Thank you for your attention!