Numerical scheme for relativistic Boltzmann equation with spherical geometry

Mikalai Prakapenia

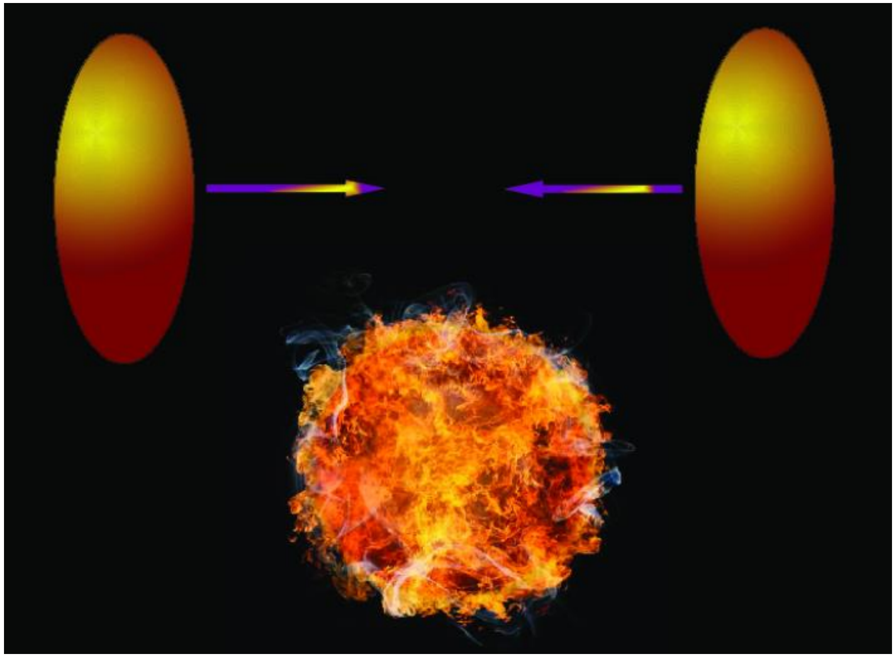
ICRANet-Minsk, Institute of physics, NASB, Minsk, Belarus

MMCP 2024

24.10.2024 Yerevan

Fireballs in nuclear physics and astrophysics

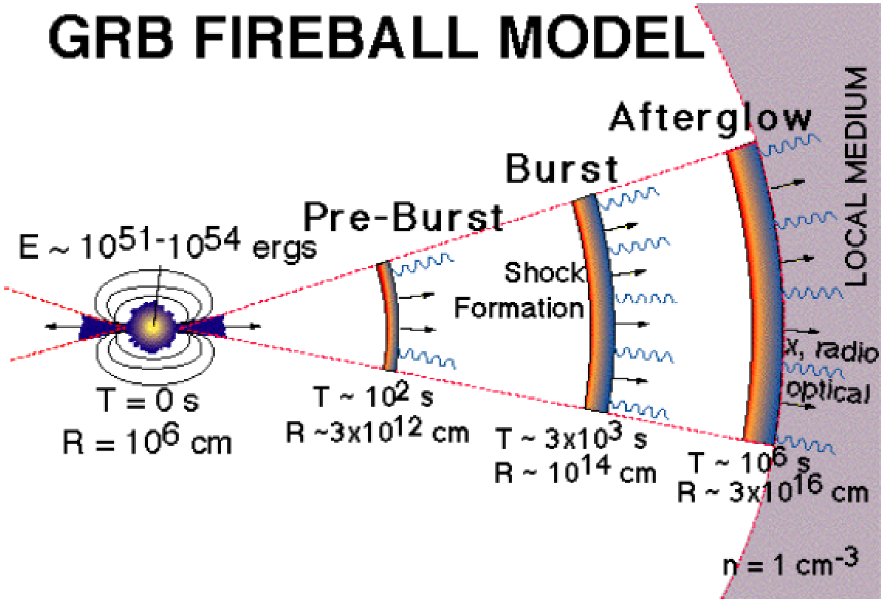
Heavy-ion collisions



Yukalov et al. Physics of Particles and Nuclei 54 (1):1-68 (2023)
Voskresensky, Nuclear Physics A 744 (2004) 378-444

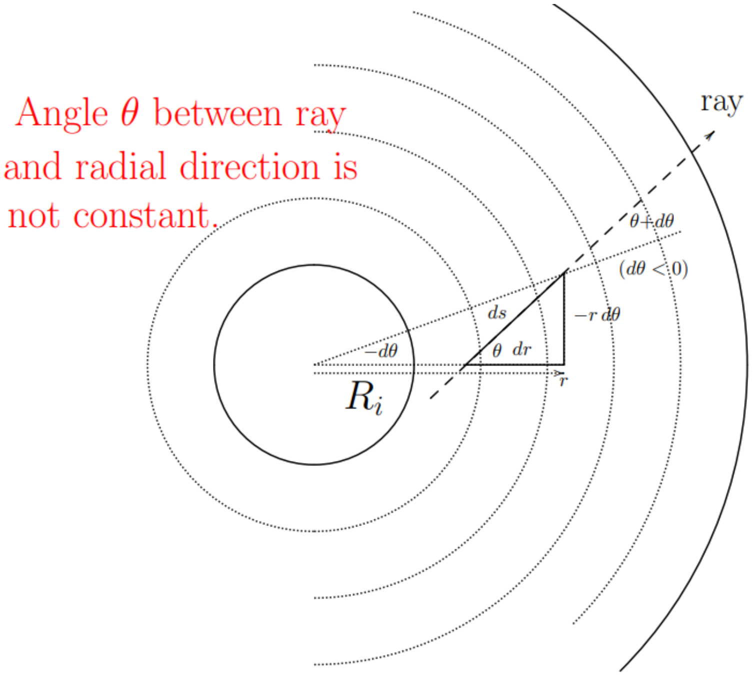
Fireballs in nuclear physics and astrophysics

Gamma-ray burst



Ghisellini, arXiv:astro-ph/0111584
Piran, Rev. Mod. Phys. 76, 1143 (2005)

Spherical Boltzmann kinetic equation



r – radial distance

p – particle momentum

$\theta = \arccos\mu = \angle(r, p)$

$$f_i(t, p, r, \mu)$$

transport term

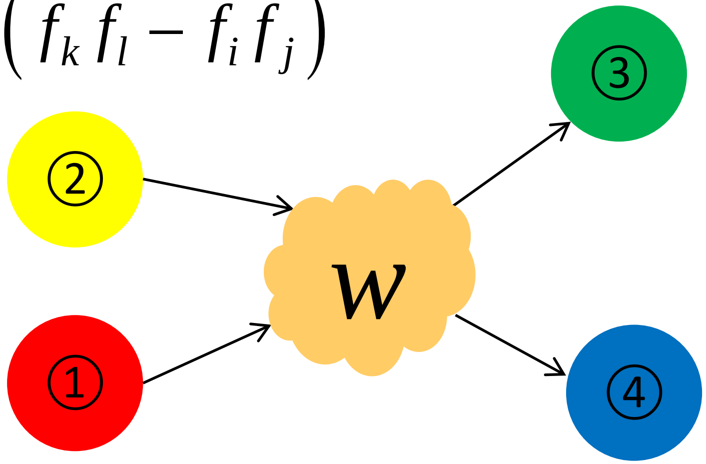
$$\frac{1}{c}\frac{\partial f_i}{\partial t} + \beta_i(\varepsilon_i)\left(\mu\frac{\partial f_i}{\partial r} + \frac{1-\mu^2}{r}\frac{\partial f_i}{\partial \mu}\right) =$$

$$\int d^3p_j d^3p_k\ d^3p_l W_{(ij|kl)}\left(f_k f_l - f_i f_j\right)$$

collisional integral

Collisional integral

$$\mathrm{St}\, f_i = \int d^3 p_j d^3 p_k \;\; d^3 p_l W_{(ij|kl)} \left(f_k f_l - f_i f_j \right)$$



$$d^3 p = p^2 dp \cdot d\mu \cdot d\varphi$$

$$p^2 = c^{-2} \varepsilon^2 - m^2 c^2$$

$$p \leftrightarrow \varepsilon$$

$$W = \frac{\hbar^2 c^6}{4\pi^2} \frac{\left| M_{if} \right|^2}{16 \varepsilon_1 \varepsilon_2 \varepsilon_3 \varepsilon_4} \delta(\varepsilon_1 + \varepsilon_2 - \varepsilon_3 - \varepsilon_4) \delta(\vec{p}_1 + \vec{p}_2 - \vec{p}_3 - \vec{p}_4)$$

- Stf is the same for any radius r

- Stf mixes energies and angles

$$(r, s, \varepsilon, \mu) = (radius,\ particle\ type,\ particle\ energy,\ particle\ angle)$$

$$4D\ array \;\; \rightarrow \;\; 1\mathrm{D\ array}$$

Collisional matrix

$$\mathrm{St}\, f_i = \int d^3 p_j d^3 p_k \; d^3 p_l W_{(ij|kl)} \left(f_k f_l - f_i f_j \right)$$

on finite grid
after integration over azimuth angle*

$$\sum_{\beta} \sum_{\gamma} C_{\alpha\beta\gamma} f_{\beta} f_{\gamma}$$

$$\alpha = 1..N_2, \quad N_2 = N_s N_{\varepsilon} N_{\mu}$$

Collisional matrix can be calculated only once before time integration

*Prakapenia, et al. Journal of Computational Physics, vol. 373, p. 533, (2018)
Prakapenia, et al. Physics of Plasmas, vol. 27, p. 113302, (2020)

From integro-differential equations with partial derivatives to ODE system

$$f_i(t, r, \varepsilon, \mu)$$

$$\frac{1}{c}\frac{\partial f_i}{\partial t} + \beta_i(\varepsilon_i)\left(\mu \frac{\partial f_i}{\partial r} + \frac{1-\mu^2}{r}\frac{\partial f_i}{\partial \mu}\right) = \mathrm{St}\, f_i$$

Second-order piecewise linear reconstruction scheme (PLM):
Mignone, Journal of Computational Physics (2014)
Fuksman, Mignone, The Astrophysical Journal Supplement Series (2019)

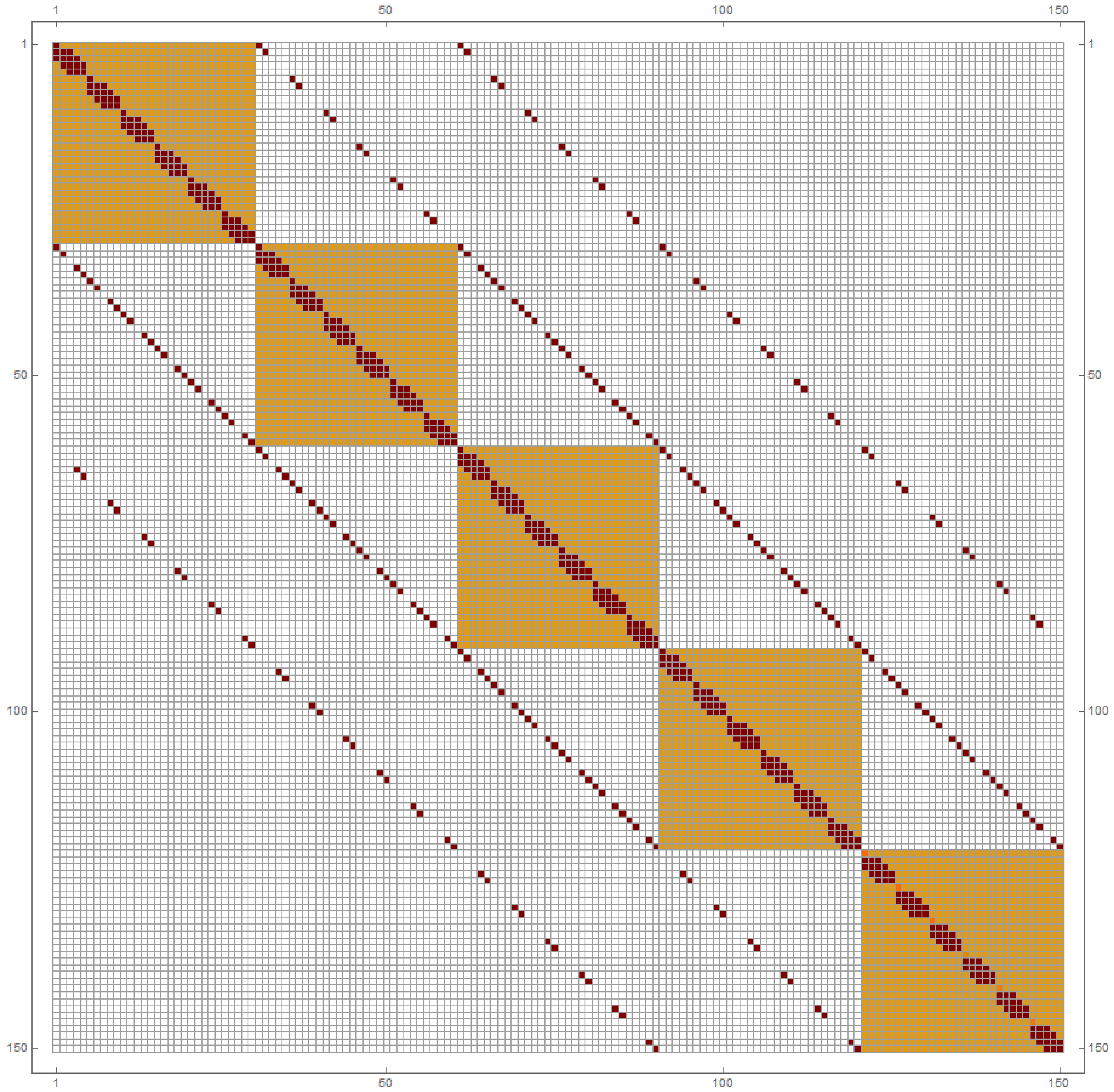
$$f_i(t, r, \varepsilon, \mu) \;\rightarrow\; f_\alpha(t)$$

$$\frac{df_\alpha}{dt} = \mathrm{F}_\alpha(f_\beta)$$

Aksenov, et al. ApJ (2004)
Bruenn, Mezzacappa , ApJ (1993)

$$\alpha = 1..N, \quad N = \underbrace{N_r}_{N_1} \cdot \underbrace{N_s N_\varepsilon N_\mu}_{N_2}$$

Jacoby matrix of ODE

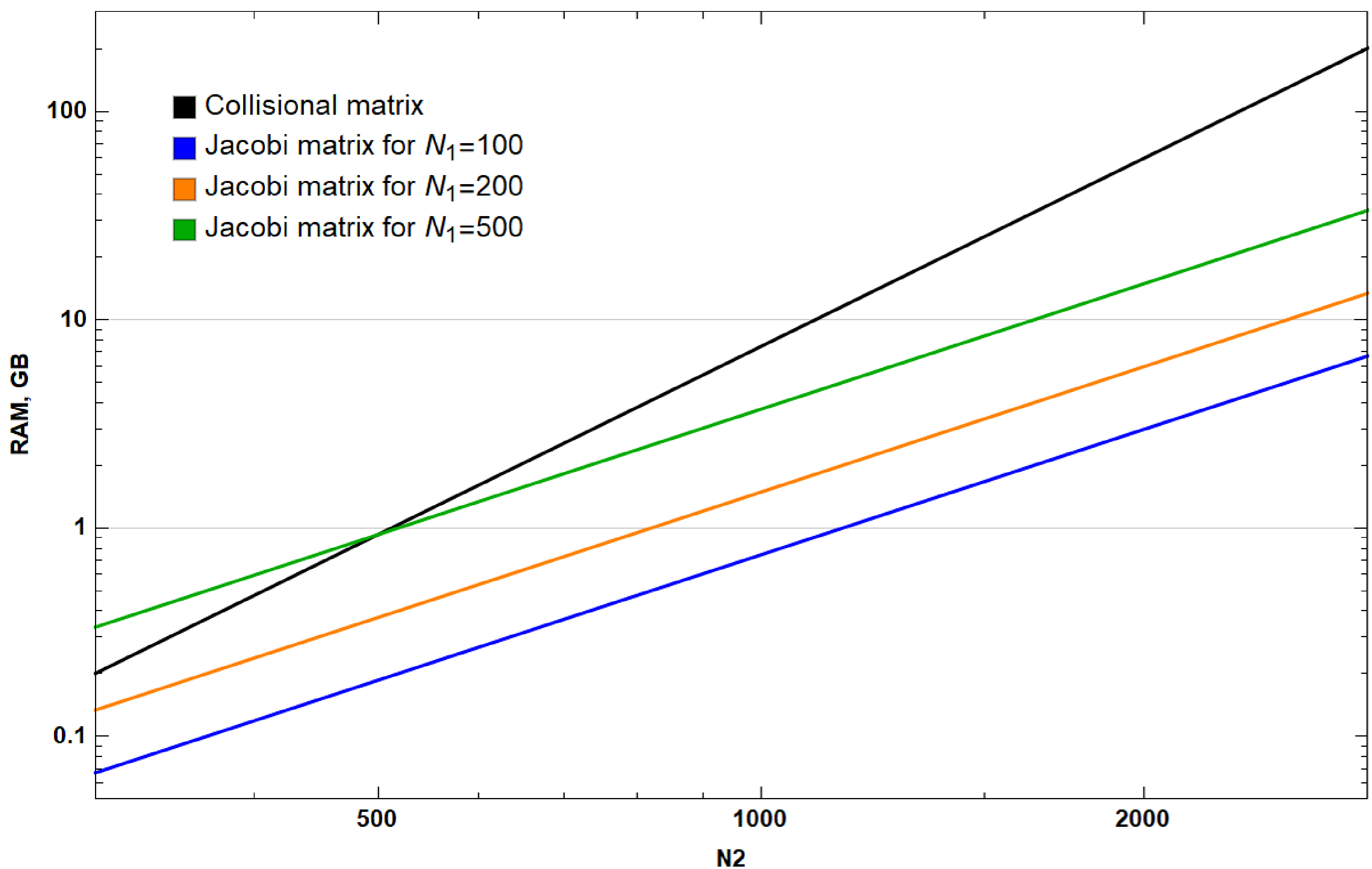


8 lines from transport

N1 blocks from Stf

Block size is N2*N2

Memory size



Collisional matrix
Jacobi matrix for N_1=100
Jacobi matrix for N_1=200
Jacobi matrix for N_1=500
100
10
1
0.1
RAM, GB
500
1000
2000
N2

Solving ODE system

$$\frac{df_\alpha}{dt} = \mathrm{F}_\alpha(f_\beta) \qquad J_{\alpha\beta} = \partial \mathrm{F}_\alpha / \partial f_\beta$$

We have a stiff system therefore we apply BDF method

Nonlinear algebraic system of equations

Newton-Raphson method with BiCGStab for SLAE

- BDF1:

  $$y_{n+1} - y_n = hf(t_{n+1}, y_{n+1})$$

  (this is the backward Euler method)
- BDF2:

  $$y_{n+2} - \frac{4}{3}y_{n+1} + \frac{1}{3}y_n = \frac{2}{3}hf(t_{n+2}, y_{n+2})$$
- BDF3:

  $$y_{n+3} - \frac{18}{11}y_{n+2} + \frac{9}{11}y_{n+1} - \frac{2}{11}y_n = \frac{6}{11}hf(t_{n+3}, y_{n+3})$$
- BDF4:

  $$y_{n+4} - \frac{48}{25}y_{n+3} + \frac{36}{25}y_{n+2} - \frac{16}{25}y_{n+1} + \frac{3}{25}y_n = \frac{12}{25}hf(t_{n+4}, y_{n+4})$$
- BDF5:

  $$y_{n+5} - \frac{300}{137}y_{n+4} + \frac{300}{137}y_{n+3} - \frac{200}{137}y_{n+2} + \frac{75}{137}y_{n+1} - \frac{12}{137}y_n = \frac{60}{137}hf(t_{n+5}, y_{n+5})$$
- BDF6:

  $$y_{n+6} - \frac{360}{147}y_{n+5} + \frac{450}{147}y_{n+4} - \frac{400}{147}y_{n+3} + \frac{225}{147}y_{n+2} - \frac{72}{147}y_{n+1} + \frac{10}{147}y_n = \frac{60}{147}hf(t_{n+6}, y_{n+6})$$

Number of operations and parallelization

At each step we have to calculate the right hand side and the Jacobi matrix of the system

$$N_1 \text{ times: } \sum_{\beta}\sum_{\gamma} C_{\alpha\beta\gamma} f_\beta f_\gamma \Rightarrow N_1 N_2^3 - operations$$

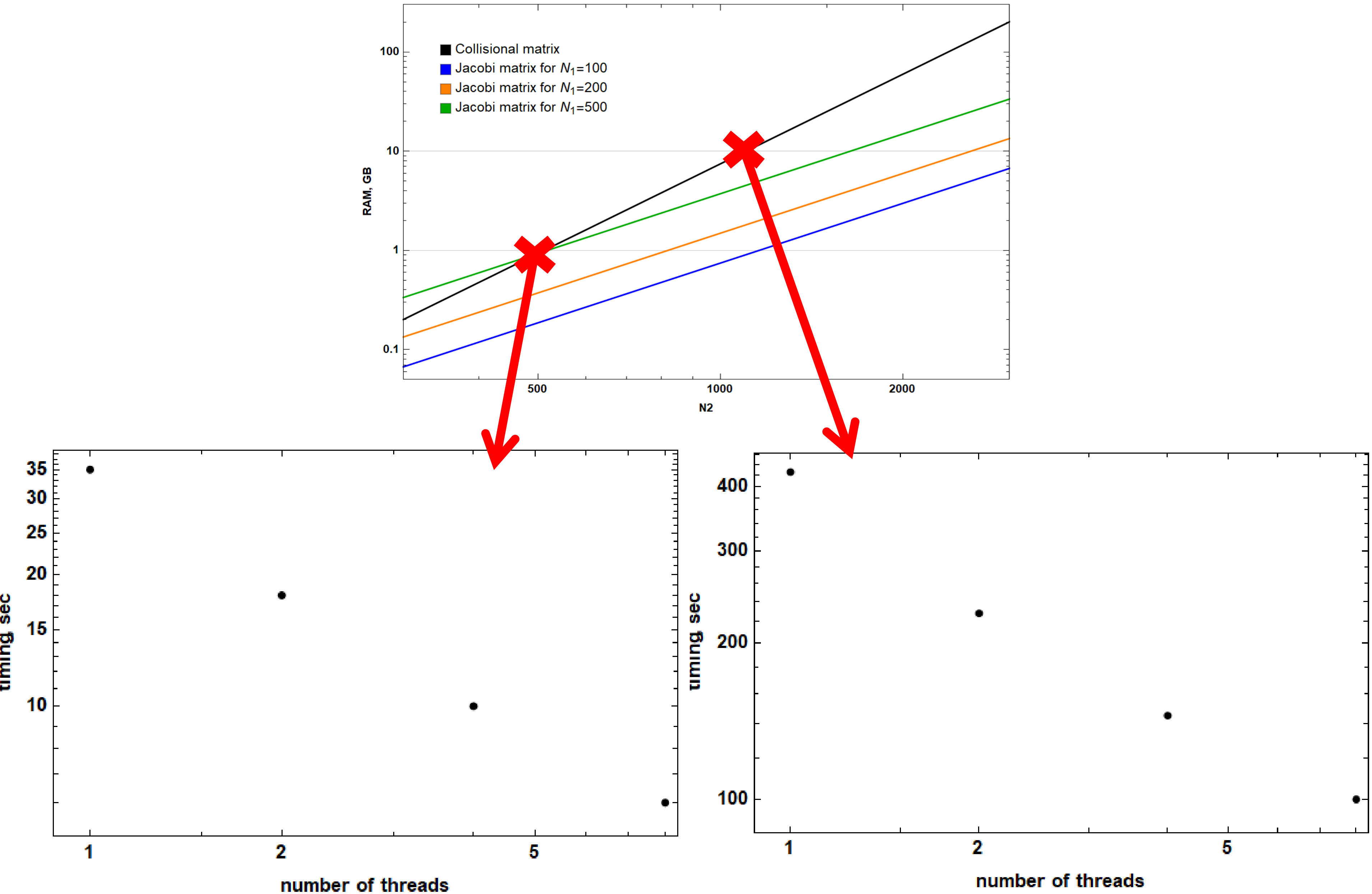
Then we solve SLAE

BiCGStab has matrix on vector product: $N_1 N_2^2 - operations$

number of iterations L << N2

Collisional integral is the bottleneck !

Numerical experiment: timing for one step



Conclusions

- **Spherical relativistic Boltzmann code for binary processes**

- **CPU OpenMP realization works well for a small grid**

- **It requires GPU realization for a larger grids**

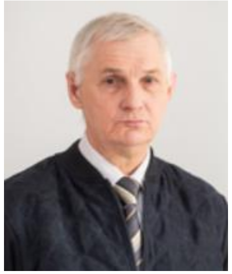
- **Possible applications:**
radiation/neutrino transfer in static/expanding medium
Thermalization during expansion of pion/quark-gluon plasma

Collaboration

Dr. Yauheni Talochka (JINR)

Dr. David Melon-Fuksman (MPIA)

Prof. Nikolai Likhoded (BSU)



Prof. Gregory Vereshchagin (ICRANet)

Prof. Dmitry Voskresensky (JINR)

Thank you