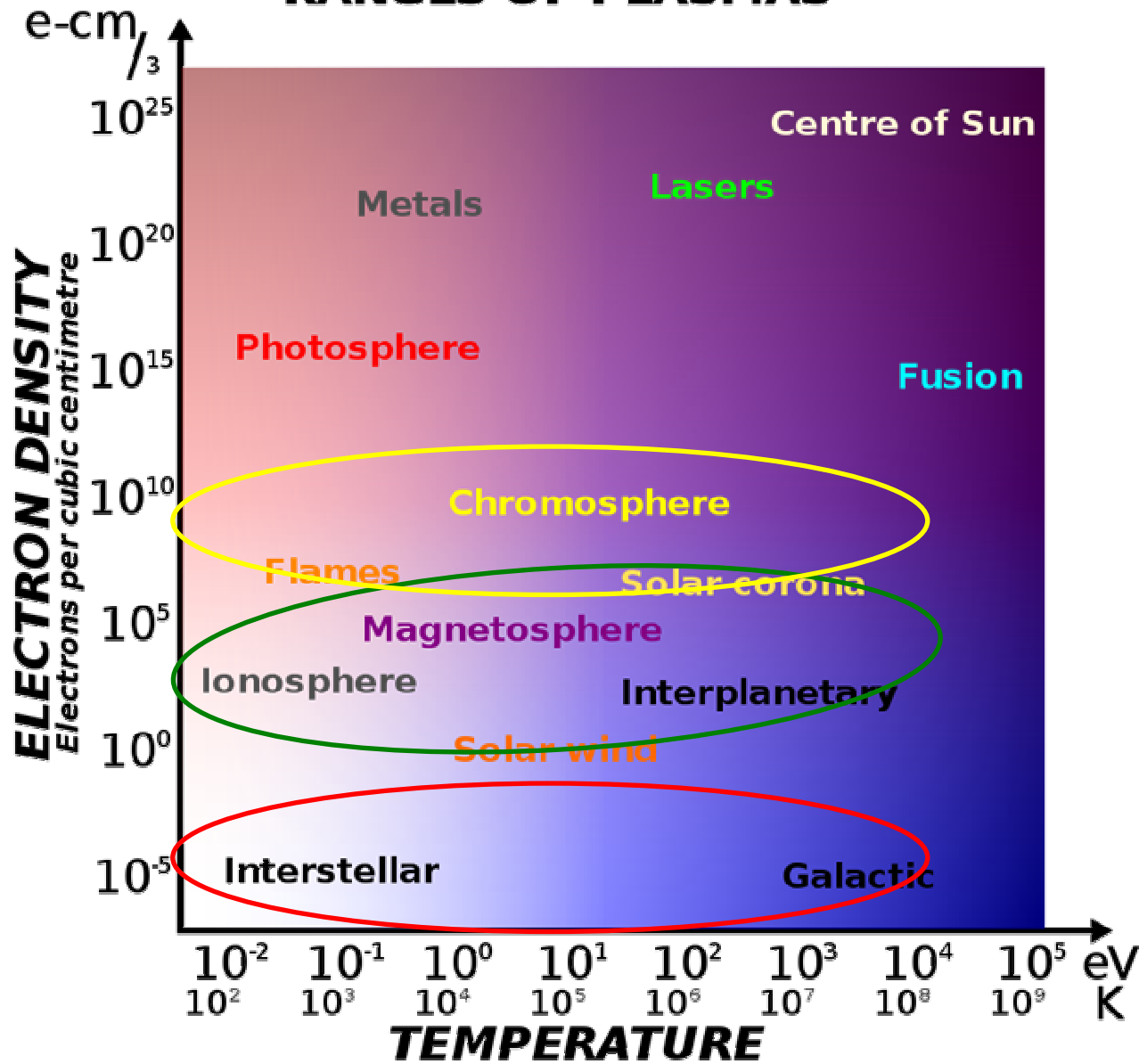


Вычислительные задачи физики космической плазмы

А.А. Петрукович, Х.В. Малова, М.С. Долгоносов

с участием О.В. Мингалева, И.В. Мингалева, ПГИ КНЦ РАН

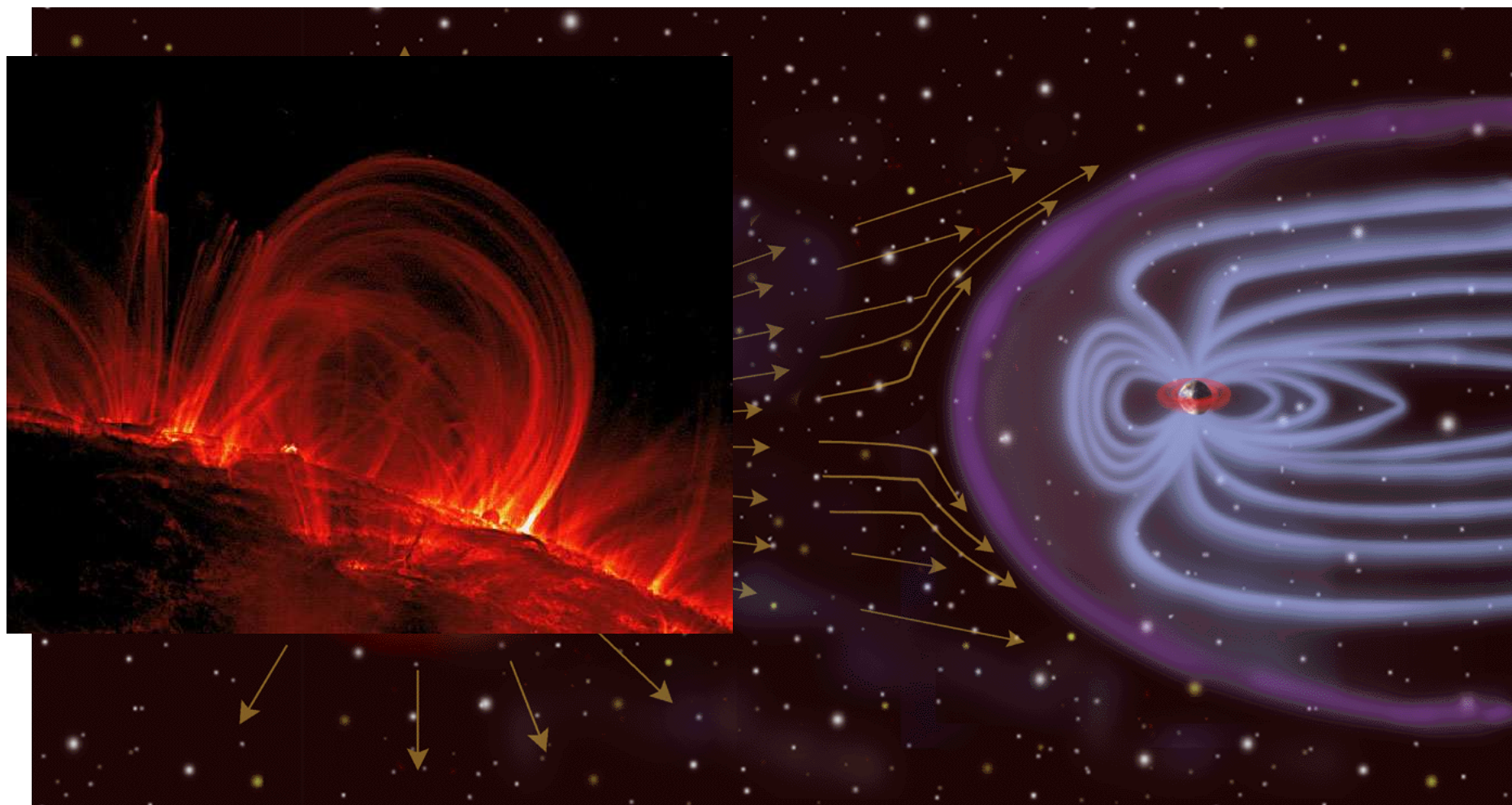
RANGES OF PLASMAS

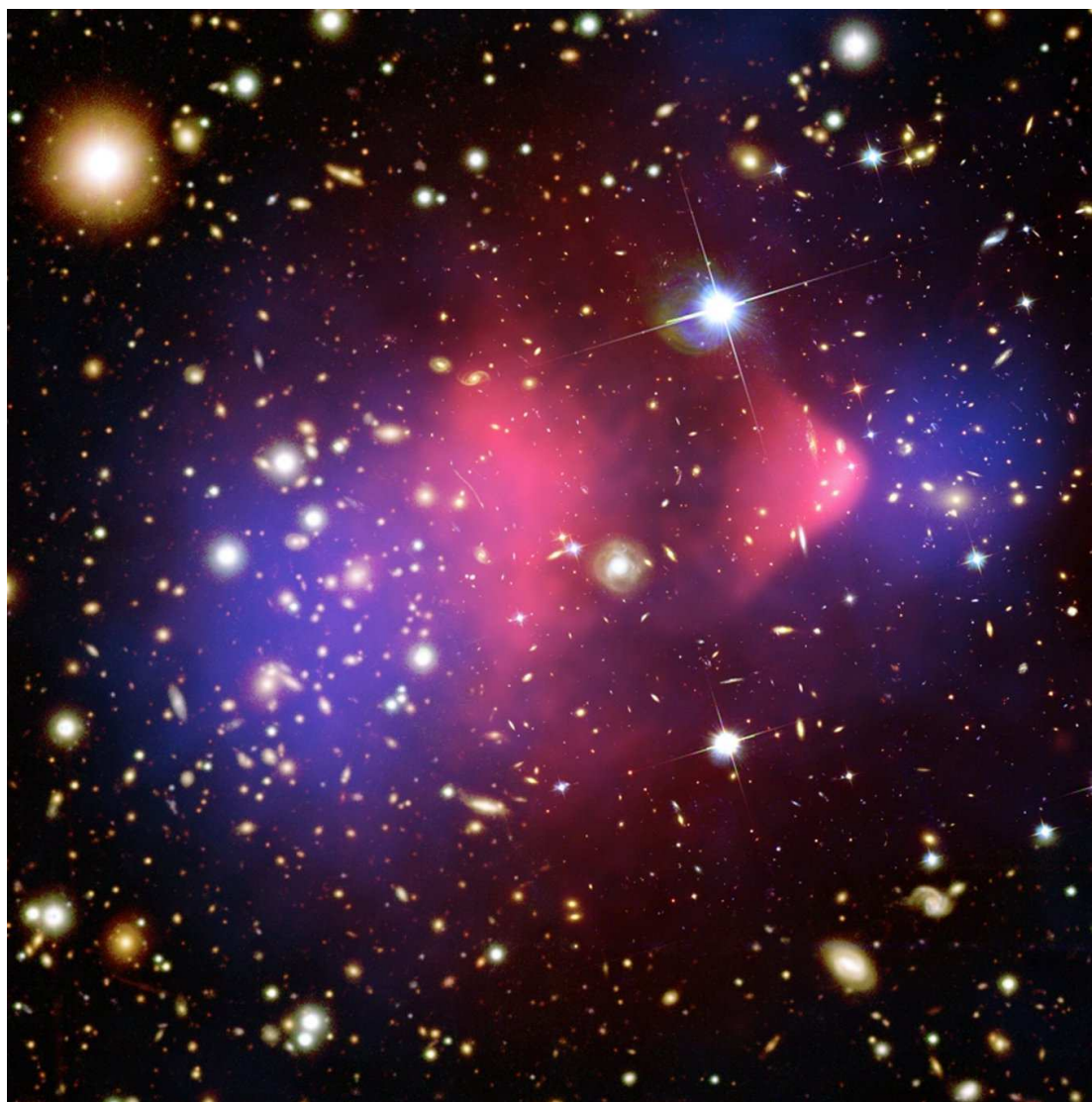


длина свободного пробега сравнима с размером системы
«бесстолкновительная» плазма

Солнечно-земные связи или космическая погода

Взаимодействие плазмы и магнитных полей
солнечного и земного происхождения





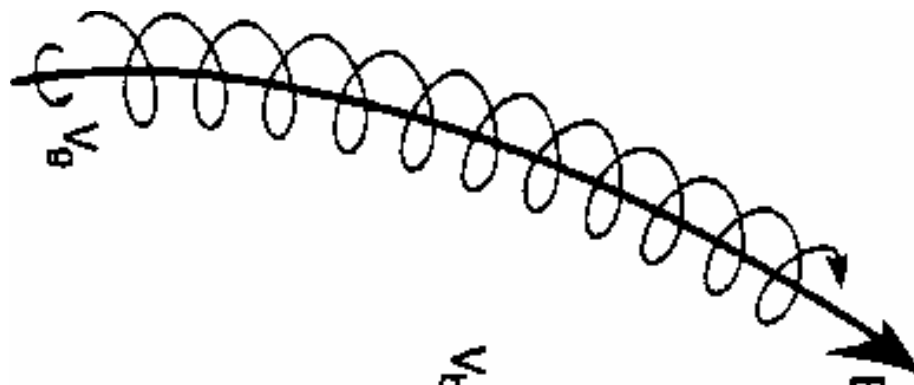
мегапарсек

Полная система уравнений: Власова-Максвелла

- Функция распределения по скоростям в каждой точке
- Три компоненты магнитного и электрического поля
- Несколько сортов частиц: масштаб для ионов и электронов отличается в 2000 раз

в общем случае плохорешаемая задача, но

В отсутствие столкновений плазма «вморожена» в магнитное поле что позволяет перейти к **магнитогидродинамической модели**

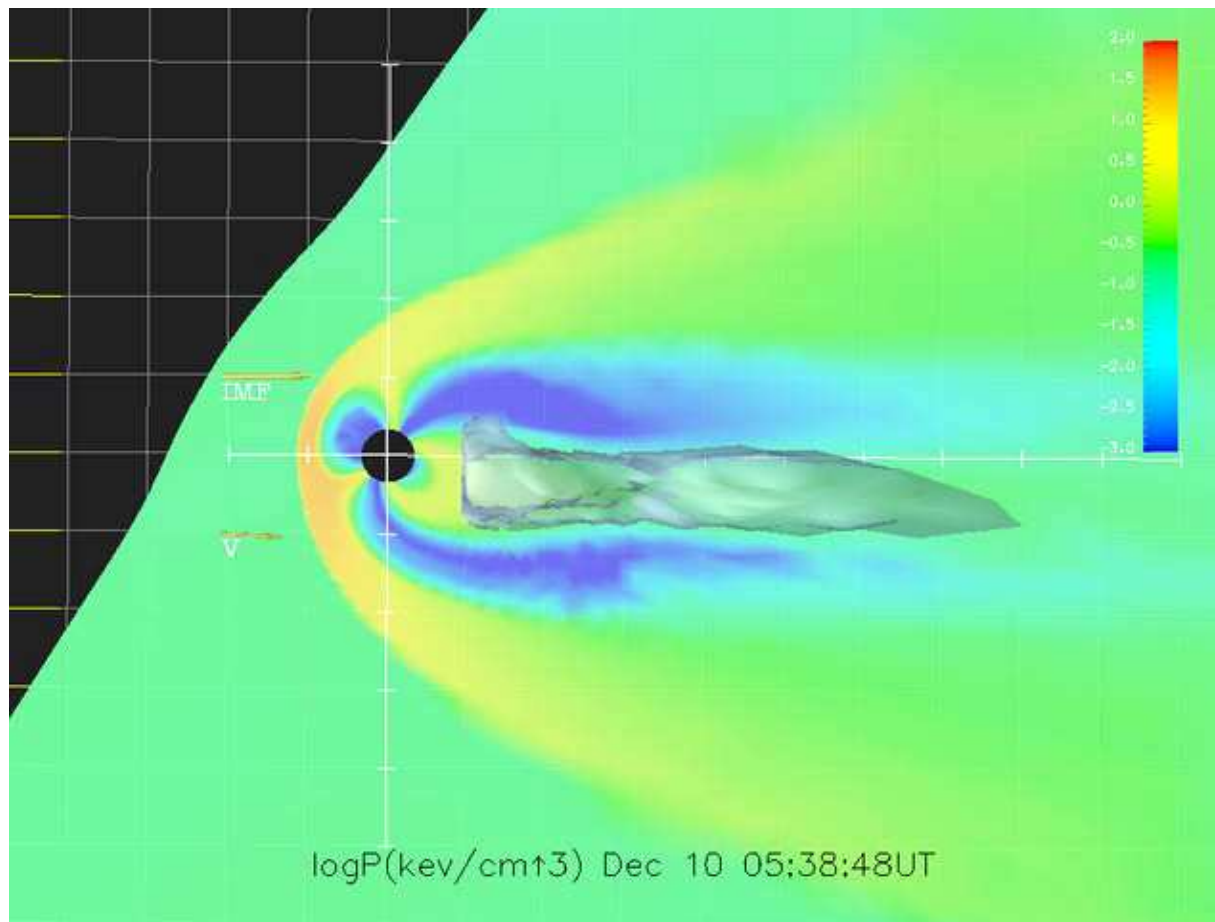


Глобальная МГД модель магнитосферы Земли

BATS-R-US: Минимум 8 вычислительных узлов, ~ 1ГБ памяти на узел

3-х мерная сетка ~100x100x100

разбивается на ячейки по числу вычислительных узлов



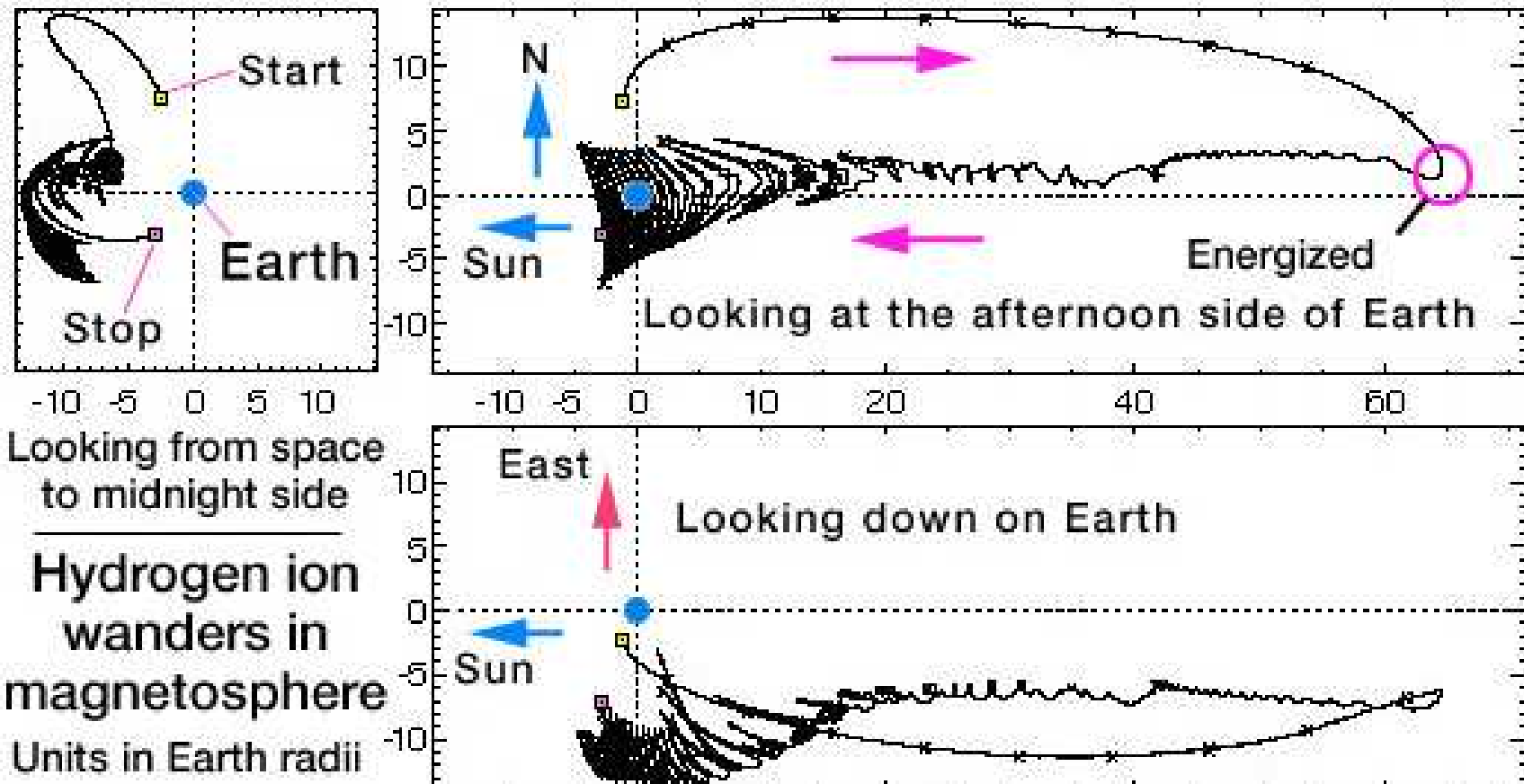
От МГД до полной кинетики:

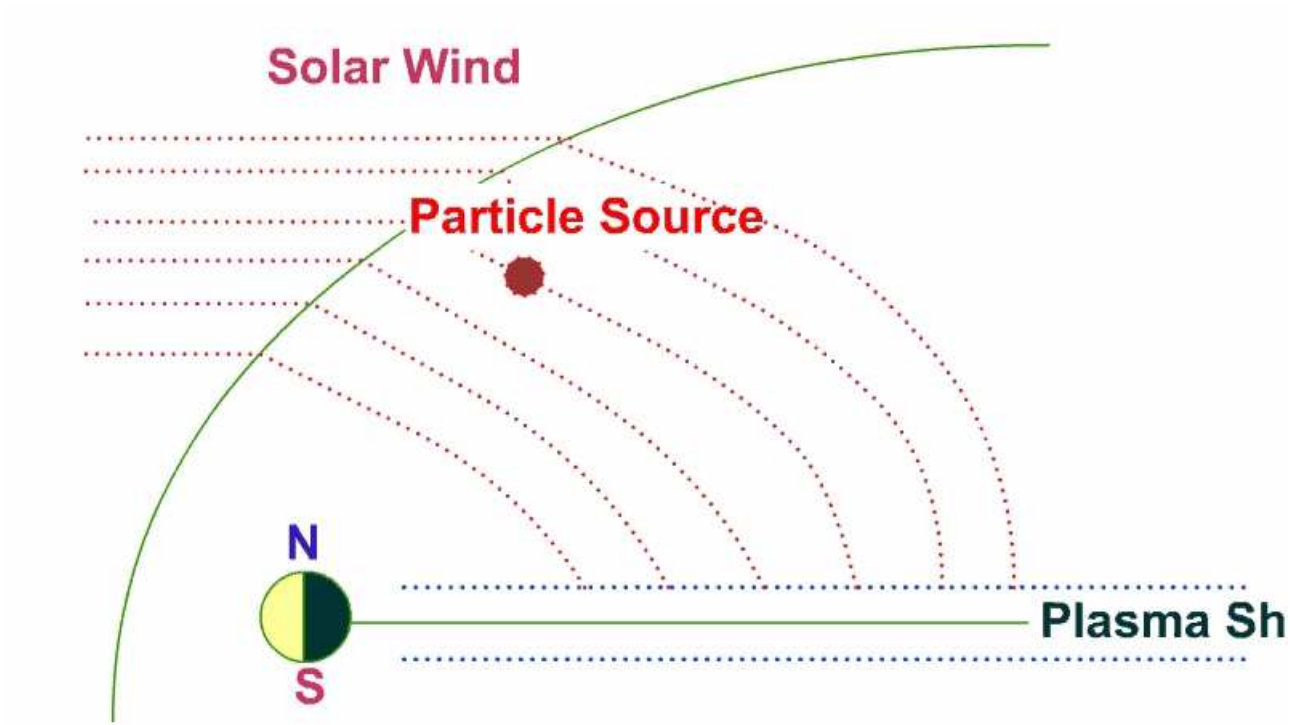
- Многожидкостная МГД
- Гибридная схема (электронная МГД плюс ионная кинетика)
- Метод крупных частиц

Моделирование методом одиночных частиц

уравнения движения в сложных конфигурациях полей

Параллелизация за счет одновременного запуска нескольких независимых расчетов





Решение систем интегро-дифференциальных уравнений

Финальный одномерный профиль собирается последовательными приближениями интегрированием по многомерному пространству

$$df/dt = 0$$

$$\vec{B} = \{B_x(x, z), B_z(x, z)\}$$

$$\frac{\partial B_x}{\partial z} - \frac{\partial B_z}{\partial x} = \frac{4\pi}{c} \left\{ \int_{V^3} v_y f_{iso}(\vec{v}, x, z) d^3v + \int_{V^3} v_y f_{aniso}(\vec{v}, x, z) d^3v + j_e(x, z) \right\}$$

$$B|_{z=L} = B_0, \varphi|_{z=L} = \left(f_{aniso}(\vec{v}) = \frac{n_{0aniso}}{(\sqrt{\pi}v_{T1})^3} \exp\left\{ -\frac{(v - V_{D1})^2 + v_{\perp}^2}{v_{T1}^2} \right\} \equiv \tilde{f}(H, I_z) \right)$$

$$f_{iso}(\vec{v}) = \frac{n_{0iso}}{(\sqrt{\pi}v_{T2})^3} \exp\left\{ \frac{P_y V_{D2} - H}{T_2} \right\} = N_0(x, z) \exp\left\{ -\frac{v_x^2 + (v_y - V_{D2})^2 + v_z^2}{v_{T2}^2} \right\};$$

$$N_0(x, z) = \frac{n_{0iso}}{(\sqrt{\pi}v_{T2})^3} \exp\left[\frac{e V_{D2} A_y(x, z) - e \varphi(x, z)}{T_2} \right]; \quad T_2 = \frac{mv_T^2}{2}$$

Практика использованию видеокарт *Nvidia* для вычислений:

- Ранее нами использовался компилятор Фортрана Portland Group 10.0 с поддержкой CUDA и видеокарты GTX 260 GTX 9800
- При умножении матриц удавалось получить ускорение в вычислениях примерно в 100 раз быстрее чем одно ядро процессора Intel Core TM2
- При дифференцировании многомерного массива по разным измерениям удавалось получить ускорение в вычислениях примерно в 40-50 раз быстрее чем одно ядро процессора Intel Core TM2
- При запуске хорошо распаралеленных газодинамических расчетов (код в Фортране большого размера) удавалось получить ускорение в вычислениях не более чем в 2 раза быстрее чем одно ядро процессора Intel Core TM2, хотя все данные помещаются в памяти видеокарты.
- Встроенные функции Фортрана $\sin(x)$, $\cos(x)$, $\exp(x)$ и другие на видеокарте работают с плохой точностью (ошибки в 2-х последних знаках) по сравнению с центральным процессором.

Вопросы:

1. Расчеты методом одиночных частиц и решение сложных систем

2. МГД и кинетические схемы

✓ Какое количество «независимых» ниток можно запустить?

✓ Какие ограничения на длину кода каждой нитки ?

Можно ли при сложных вычислениях (большая длина кода) добиться ускорения такого же как при простых вычислениях? Если можно, то как?

✓ Какие ограничения на хранение и обмен данными ?

Размеры кэша для команд и данных каждой нити для видеокарт новой архитектуры ФЕРМИ ?

Требования к программному обеспечению компьютера

- Наличие Windows XP SP 3 (32 bit)
- Visual Studio 2008
- Компилятор Фортрана PGI Visual Fortran 10.0;
- компилятор Cpp
- Драйверы и программы для Cuda и компилятор NVCC для Windows
- Intel Visual Fortran 11.1.051