



Equilibrium–Stability application within MAPPER infrastructure



Olivier Hoenen

on behalf of the Fusion Applications group*

* Max-Planck IPP, Garching, Germany – Chalmers University, Gothenburg, Sweden.

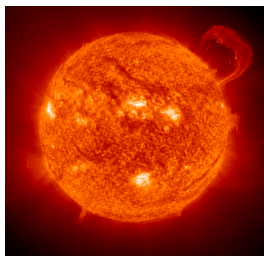


Summer School on Distributed Multiscale Computing, 03.06.2013

Introduction to Fusion



- Fusion reaction
 - thermo-nuclear Fusion is a potential source of energy
 - require a large enough product $nT\tau$
 - 4th state of matter: plasma
- Magnetic confinement
 - in Tokamak (ITER) or Stellarator
 - challenge: confine hot plasmas on longer times
 - problem: turbulence and instabilities degrade plasma confinement
 - need simulation to better understand the plasma, and develop optimal scenario in fusion devices

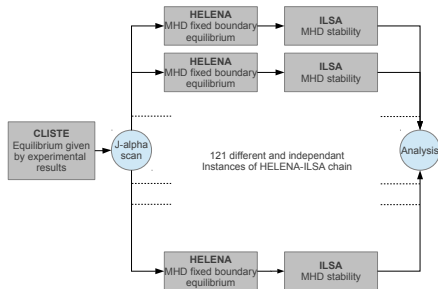




- Study of plasma as a conducting fluid → *Magnetohydrodynamics*
- Importance of stability:
 - Instability → confinement degradation → disruption (sudden loss of thermal energy → end of discharge)
- Various types of instabilities (hydrodynamic and kinetic)
- Improving stability
 - design and configuration of the Fusion device
 - control internal structure of the plasma through external heating and current drive

1. Equilibrium reconstruction
 - from experimental shot
 - free boundary codes
2. Equilibrium refinement
 - high resolution equilibrium
 - fixed boundary codes
3. MHD stability analysis
 - effects of small perturbations

- J - α scan: different equilibrium profiles, variation of pressure and current density



Loosely-coupled simulation



- Acyclic multi-physics/models case
- Two submodels:
 - HELENA (fixed boundary equilibrium)
 - ILSA (linear MHD stability)
 - exchange of data through files
- Visualization with Python/Matplotlib
- Simulation designed and ran using MAPPER online tools:
 1. MaMe: register all submodels (HELENA, ILSA as executables)
 2. MAD: design the dataflow between involved models
 3. GridSpace: execute each part on targeted machine



- Each model is following ITM's guidelines:
 - I/O made of standardized physics objects: CPO
 - equilibrium: input for HELENA/ILSA, output for HELENA
 - mhd: output for ILSA
 - Codes solving the same physics are sharing the same interface
 - Code specific parameters given in XML
- Advantage: once the *workflow* is built, changing the code solving a given part of the physics is transparent
 - ease benchmarking
 - allow simulations involving lots of models/physics
- Drawback: overhead in development and usage of CPO



Simple Equilibrium-Stability chain