Real-time simulation of the NBI fast-ion distribution

M. Weiland[1], R. Bilato[1], R. Dux[1], B. Geiger[1], A. Lebschy[1],
C.S. Colins[2], E. Fable[1], F. Felici[3,4], R. Fischer[1], C. Galperti[3], L. Giannone[1],
W.W. Heidbrink[5], W. Hu[6], D. Lui[5], D. Rittich[1], M. Scheffer[3], B. Sieglin[1], W. Treutterer[1],
M.A. van Zeeland[5], the ASDEX Upgrade, DIII-D and MST1 teams

[1]Max-Planck-Institut für Plasmaphysik, Garching, Germany
[2]General Atomics, San Diego, California, USA
[3]Technische Universiteit Eindhoven, Netherlands
[4]École polytechnique fédérale de Lausanne, Switzerland
[5]Department of Physics and Astronomy, University of California, Irvine, California, USA
[6]Institute of Plasma Physics, Chinese Academy of Sciences, Hefei, China

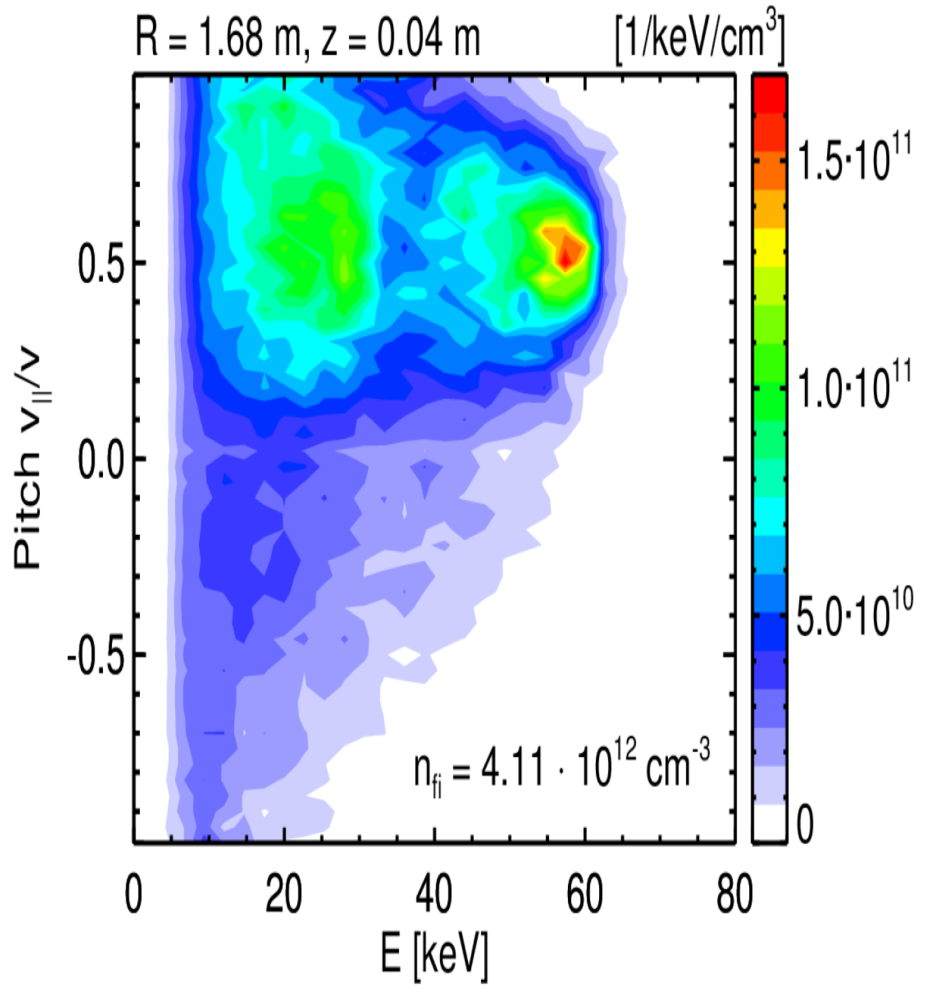
This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement number 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

RABBIT: Real-time model for the NBI fast-ion distribution

Motivation:

- Fast ion distribution function is required for instance:
  - Heating profiles for transport calculations
  - pressure and current-drive for equilibrium reconstructions
- Sophisticated simulation codes exist (e.g. TRANSP/NUBEAM based on Monte Carlo), but long computation time (~ 30 s per time-step)
- → Too slow for real-time applications (e.g. discharge control systems, real-time transport solvers like RAPTOR)
- → Develop fast model Rapid Analytical Based Beam Injection Tool – RABBIT [M. Weiland, NF 2018] (~ 20 ms per time-step)

TRANSP fast-ion distribution function



Kinetic equation

Kinetic equation for distribution function f(x, v, t)

$$\frac{\partial f}{\partial t} + \underbrace{\boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} f + \boldsymbol{a} \cdot \nabla_{\boldsymbol{v}} f}_{\text{Orbit effects}} = \nabla_{\boldsymbol{v}} \cdot \Gamma_c(f) \quad + \text{Source}$$

Orbit effects

$$\boldsymbol{a} = \frac{q}{m}\left(E + \frac{\boldsymbol{v}}{c} \times B\right)$$

Time dependence
(=0 for steady state
solution)

collisions
(e.g. slowing down,
pitch angle scattering)

Source = NBI
deposition

Kinetic equation - outline

Kinetic equation for distribution function f(x, v, t)

$$\frac{\partial f}{\partial t} + \underbrace{\boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} f + \boldsymbol{a} \cdot \nabla_{\boldsymbol{v}} f}_{} = \nabla_{\boldsymbol{v}} \cdot \Gamma_c(f) \quad + \text{Source}$$

3. Orbit effects

$$\boldsymbol{a} = \frac{q}{m}\left(E + \frac{\boldsymbol{v}}{c} \times B\right)$$

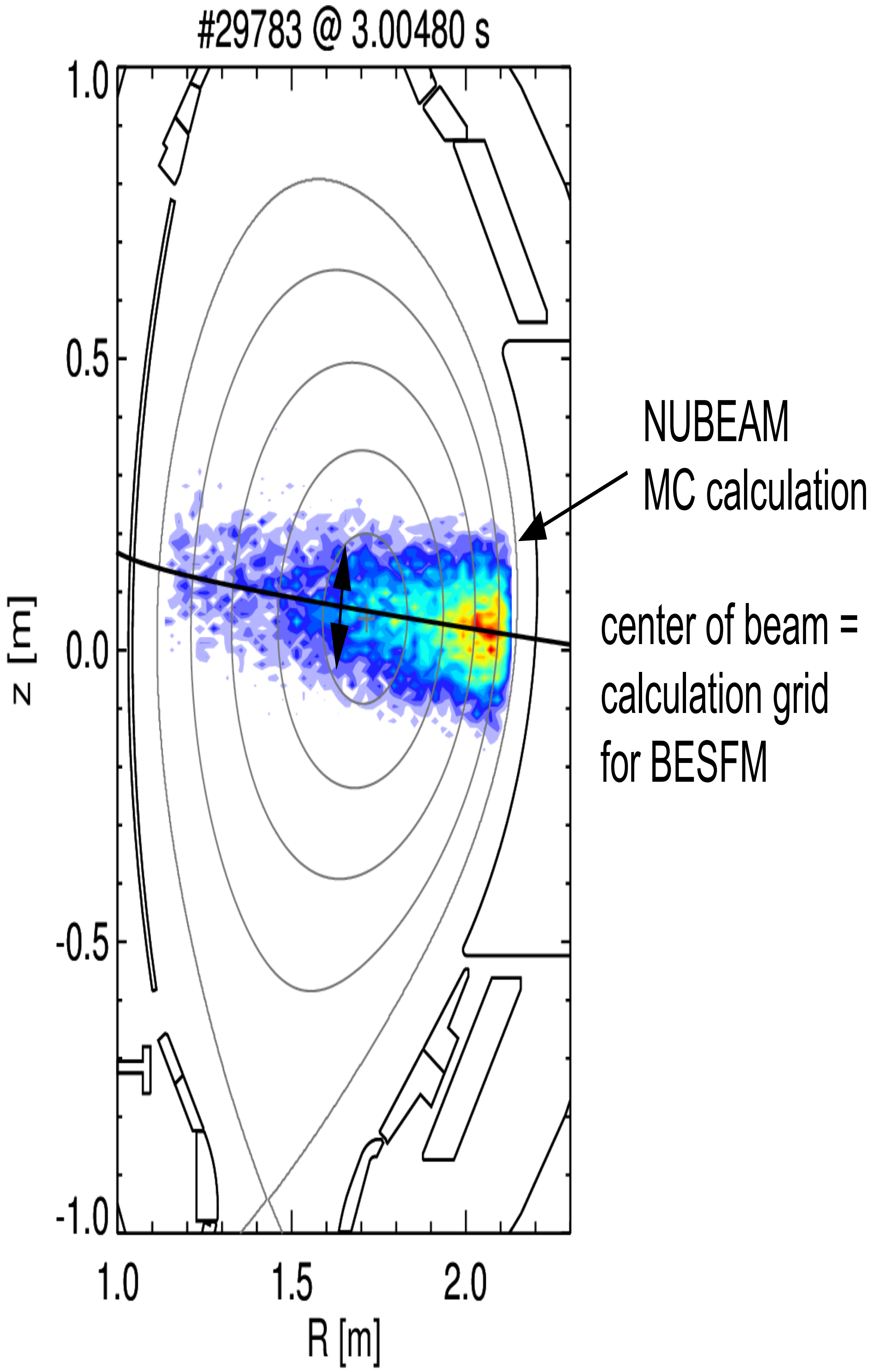
4. Time dependence
(=0 for steady state solution)

2. collisions
(e.g. slowing down, pitch angle scattering)

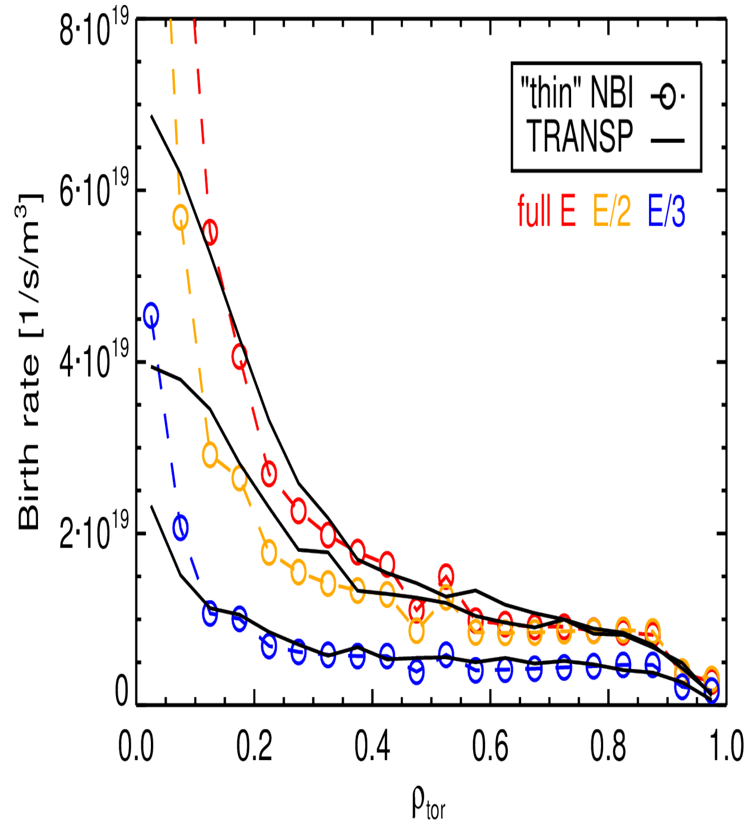
1. Source = NBI deposition

5. Applications

Beam deposition (birth profile)

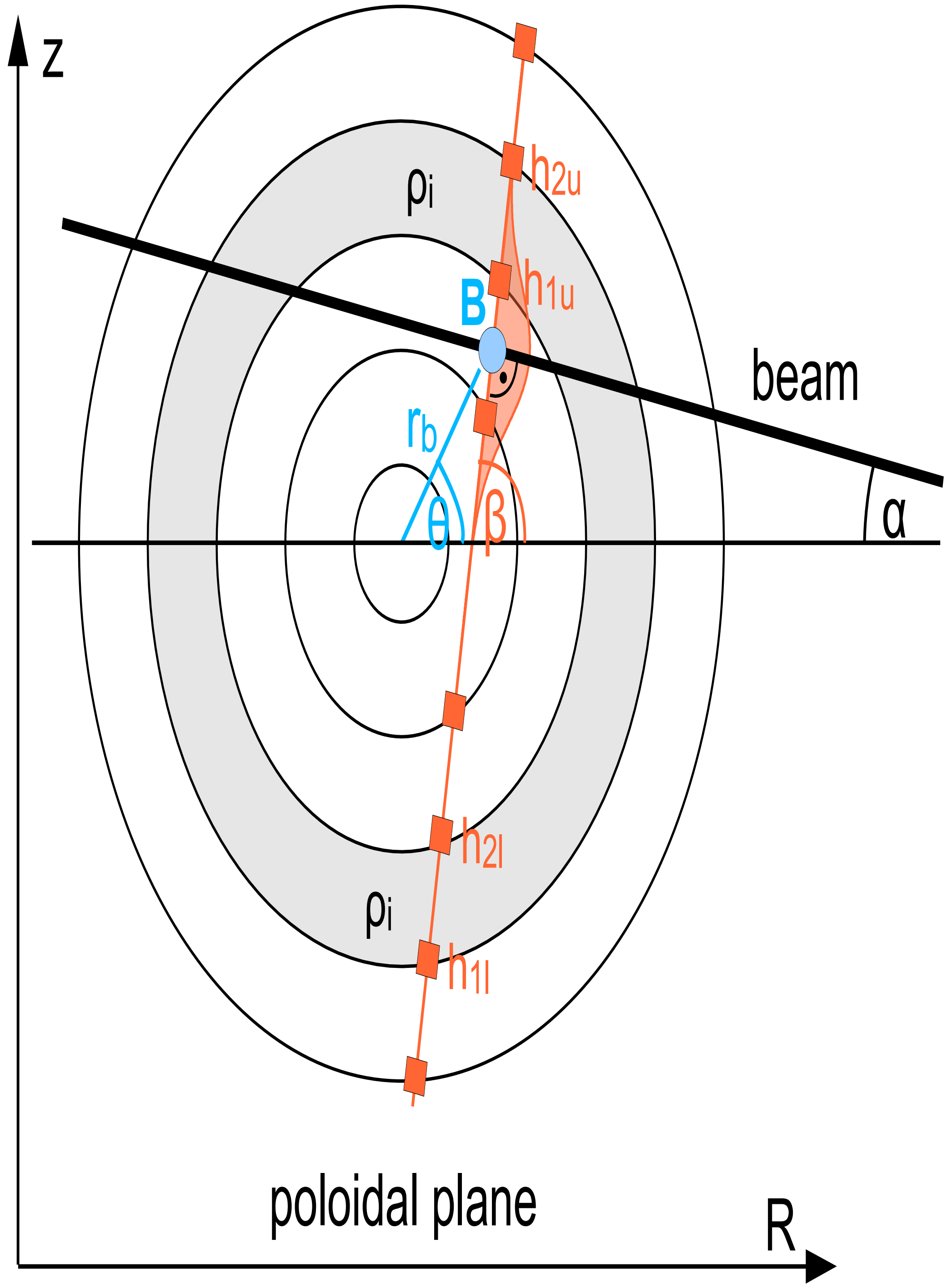


- Injection of fast neutrals into the plasma → Ionization → newly "born" fast ion
- Fast-ion birth rate = - (beam attenuation rate)
- Calculation of beam attenuation: BESFM Code by A. Lebschy, R. Dux, IPP
- We use the simplest geometry: NBI as thin line
- Good approximation for attenuation – for birth profile, we need to take into account the beam width:



- Assume a Gaussian broadening with standard deviation σ(l), l = coordinate along beam, σ(l) defined by NBI parameters (e.g. divergence)

Analytic model for the poloidal beam width



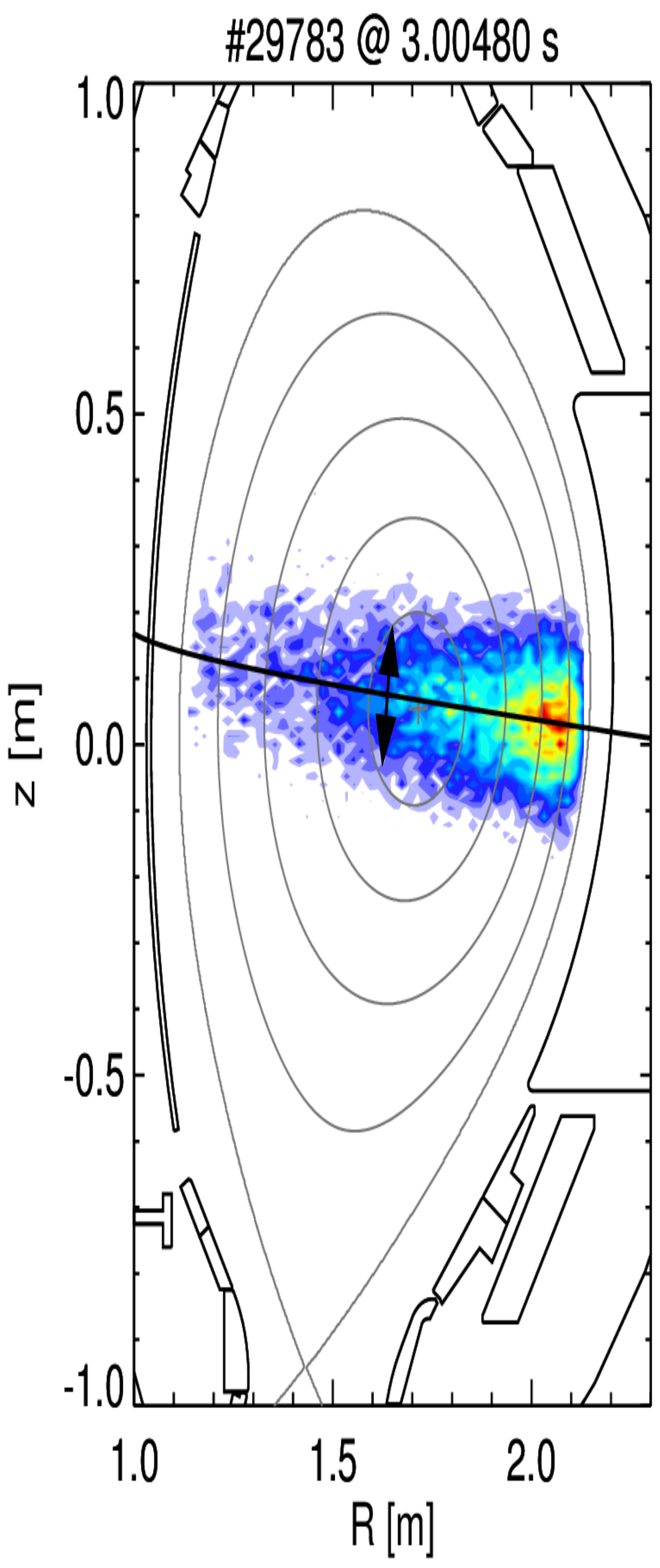
- We assume Gaussian spreading along orange line (standard deviation σ)
- Assume circular concentric flux surfaces
- Transformation between flux coordinate ρ and geometric radius r based on ratio at B:
  $r(\rho) = \rho * (r_b / \rho_b)$
- → Crossings ■ with ρ-cells can be calculated analytically
- → Contribution into i-th cell ρ_i:

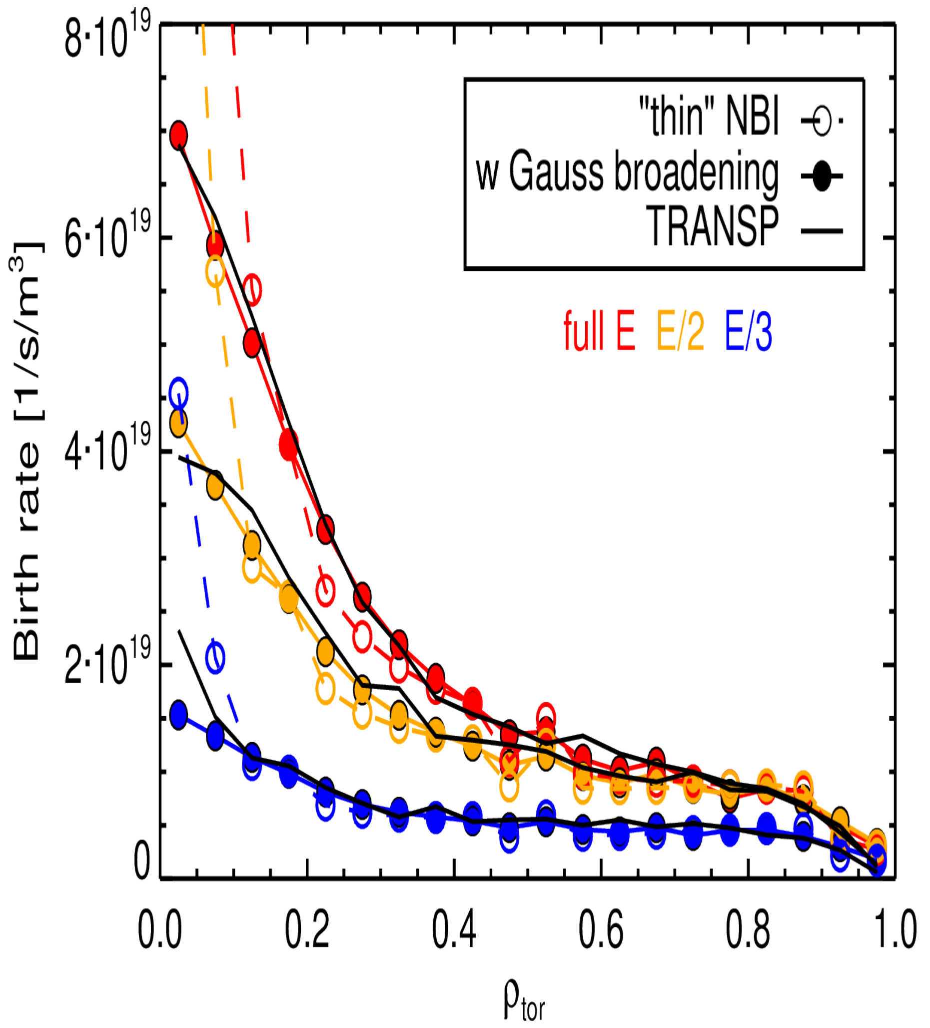
$$\frac{1}{2}\left(\operatorname{erf}\frac{h_{2u}}{\sqrt{2}\sigma} - \operatorname{erf}\frac{h_{1u}}{\sqrt{2}\sigma}\right) + \frac{1}{2}\left(\operatorname{erf}\frac{h_{2l}}{\sqrt{2}\sigma} - \operatorname{erf}\frac{h_{1l}}{\sqrt{2}\sigma}\right)$$

- Correction for plasma elongation:
  Scale beam width σ according to elongation b/a at B.

$$\sigma = \sigma_0 \cdot \sqrt{\frac{(a\cos\theta)^2 + (b\sin\theta)^2}{(a\cos\beta)^2 + (b\sin\beta)^2}}$$

Beam deposition (birth profile) with beam-width correction





- Taking into account a Gaussian broadening of the beam leads to good agreement with TRANSP/NUBEAM

Kinetic equation - outline

Kinetic equation for distribution function f(x, v, t)

$$\frac{\partial f}{\partial t} + \underbrace{\cancel{\boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} f + \boldsymbol{a} \cdot \nabla_{\boldsymbol{v}} f}}_{\text{3. Orbit effects}} = \nabla_{\boldsymbol{v}} \cdot \boldsymbol{\Gamma}_c(f) \quad + \text{Source}$$

3. Orbit effects

$$\boldsymbol{a} = \frac{q}{m}\left(\boldsymbol{E} + \frac{\boldsymbol{v}}{c} \times \boldsymbol{B}\right)$$

4. Time dependence
(=0 for steady state
solution)

2. collisions
(e.g. slowing down,
pitch angle scattering)

1. Source = NBI
deposition

5. Applications

Analytic solution of the Fokker-Planck equation

$$\underbrace{\frac{1}{\tau_s v^2}\frac{\partial}{\partial v}[(v^3+v_c^3)f]}_{\text{slowing down}} + \underbrace{\frac{\beta\, v_c^3}{\tau_s v^3}\frac{\partial}{\partial \xi}(1-\xi^2)\frac{\partial f}{\partial \xi}}_{\text{pitch angle scattering}} + \underbrace{\cancel{\frac{1}{\tau_s v^2}\frac{\partial}{\partial v}\left[\left(\frac{T_e}{m_{fi}}v^2+\frac{T_i}{m_{fi}}\frac{v_c^3}{v}\right)\frac{\partial f}{\partial v}\right]}}_{\text{speed diffusion}} = \cancel{\frac{\partial f}{\partial t}} - \underbrace{\frac{S}{2\pi v^2}\delta(v-v_0)K(\xi)}_{\text{source term}}$$

S: deposition, v0: injection velocity (mono-energetic)
K(ξ): broad pitch distribution

- Solution [Cordey/Core 74]:

Legendre polynomials

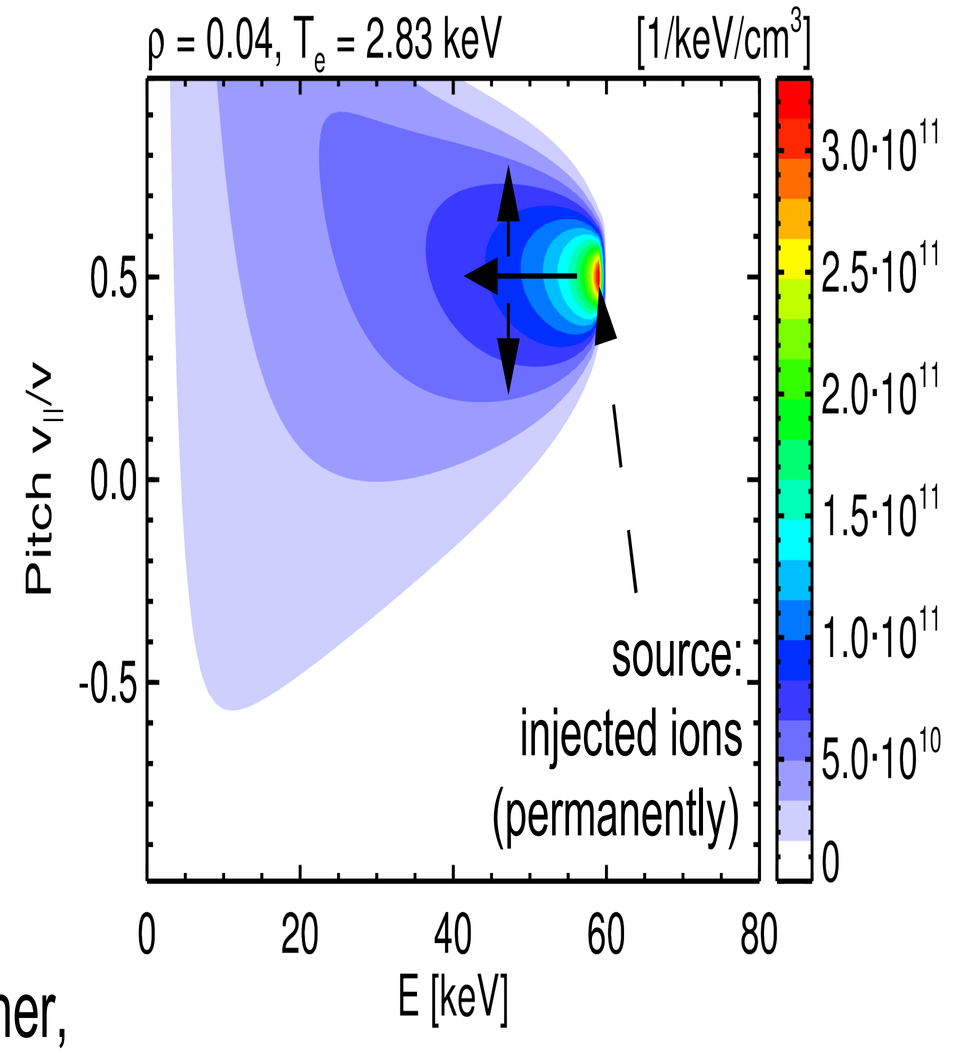
$$\boxed{f(v,\xi) = \frac{1}{2\pi}\frac{S\cdot\tau_S}{v^3+v_C^3}\cdot\sum_{l=0}^{\infty}\left(l+\tfrac{1}{2}\right)u^{l(l+1)}P_l(\xi)K_l\cdot H(v_0-v)}$$

v_C, v_0 : critical, injection velocity

$$u = \left(\frac{v_0^3+v_C^3}{v^3+v_C^3}\frac{v^3}{v_0^3}\right)^{\pm\beta/3} \qquad \beta = \frac{Z_{eff}}{2}\frac{m_i}{m_{beam}}$$

$$K_l = \int K(\xi)P_l(\xi)\,d\xi$$

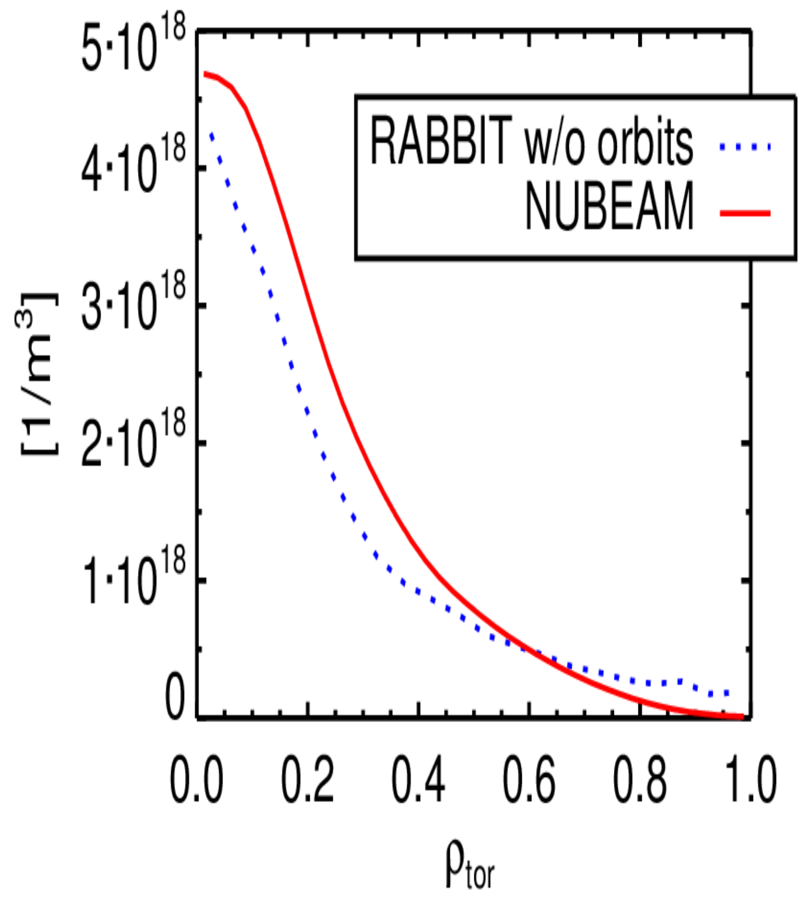
$$\tau_S = 6.27\cdot 10^8\cdot\frac{A_b}{Z_b^2\ln\Lambda_e}\cdot\frac{(T_e[\mathrm{eV}])^{3/2}}{n_e[\mathrm{cm}^{-3}]}\ \mathrm{s} \qquad v_C^3 = 1.51\cdot 10^{14}\cdot(T_e[\mathrm{eV}])^{3/2}\cdot\frac{\sum_i n_i Z_i^2\ln\Lambda_i/A_i}{n_e\ln\Lambda_e}$$

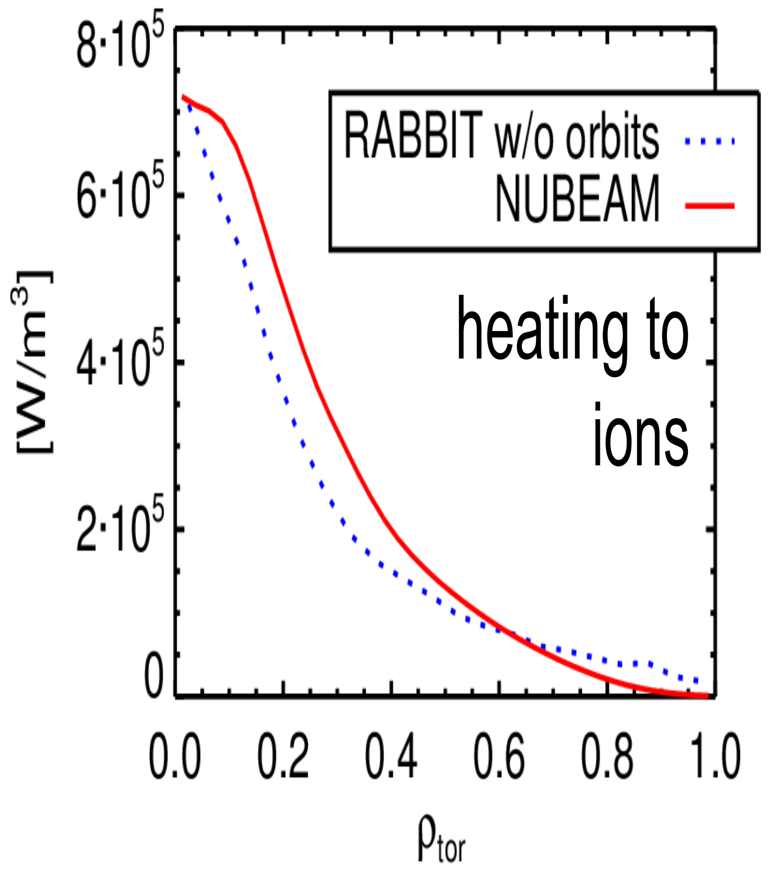


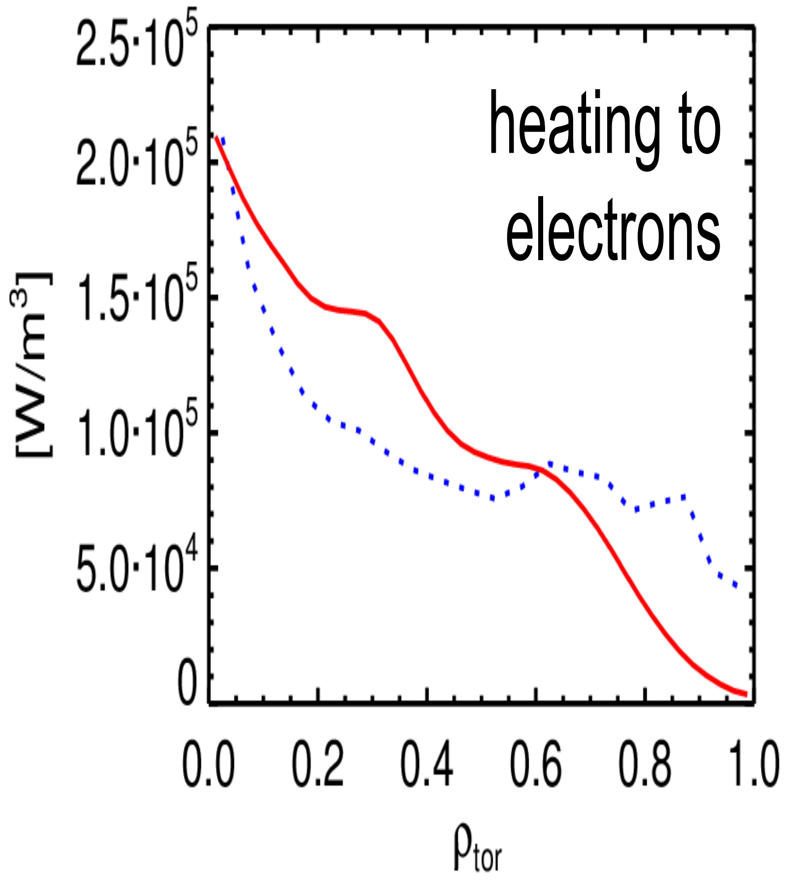
- Uniform plasma solution, i.e. each radial cell is independent of each other, no particle trapping etc.
- A correction for speed diffusion is applied above injection energy.

Density and heating profiles

- In the end we are interested in integrals of f, e.g.:
  Heating power (to electrons and ions), fast-ion pressure and current drive
- These integrals can also be solved analytically. Due to orthogonality of the Legendre polynomials, only first few moments are necessary (l=0, 1)
- E.g. fast-ion density: $n_{\mathrm{fi}} = \iint 2\pi v^2 \cdot f(v,\xi)\,\mathrm{d}v\,\mathrm{d}\xi = \frac{S\tau_S}{3} \ln\left(\frac{v_0^3 + v_C^3}{v_C^3}\right)$







- Profile shapes do not (yet) agree well, due to missing orbit-effects
- Under-estimation in the core, over-estimation at the edge

Kinetic equation - outline

Kinetic equation for distribution function f(x, v, t)

$$\frac{\partial f}{\partial t} + \underbrace{\boldsymbol{v}\cdot\nabla_{\boldsymbol{x}} f + \boldsymbol{a}\cdot\nabla_{\boldsymbol{v}} f}_{\text{3. Orbit effects}} = \nabla_{\boldsymbol{v}}\cdot\Gamma_c(f) \quad + \text{Source}$$

3. Orbit effects

$$\boldsymbol{a} = \frac{q}{m}\left(\boldsymbol{E} + \frac{\boldsymbol{v}}{c}\times\boldsymbol{B}\right)$$

4. Time dependence (=0 for steady state solution)

2. collisions (e.g. slowing down, pitch angle scattering)

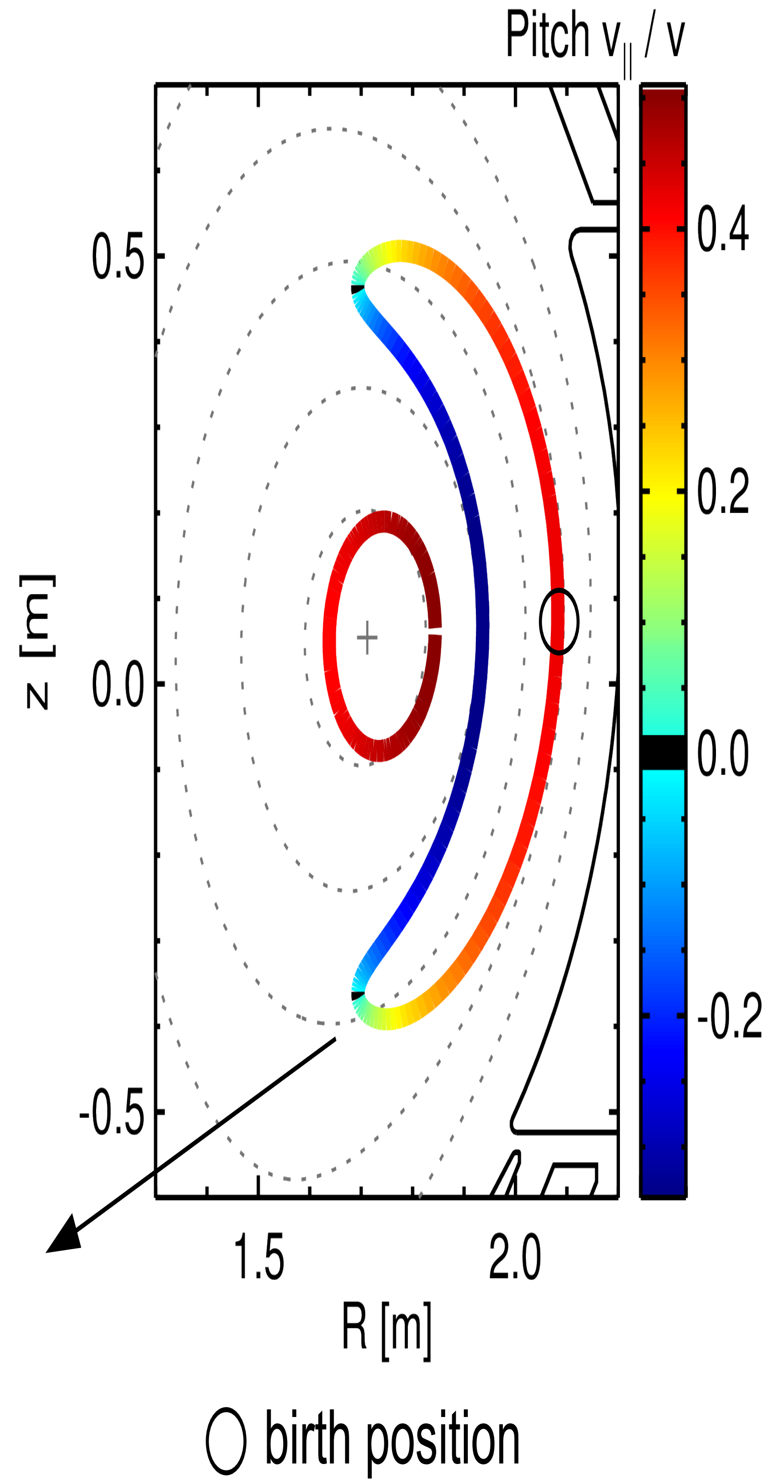
1. Source = NBI deposition

- In MC codes (e.g. NUBEAM)
  - MC representation of source
  - Calculate orbits for each MC marker
  - Apply collision operator during orbit steps
- For real-time: Only ad-hoc treatment of orbit effects possible

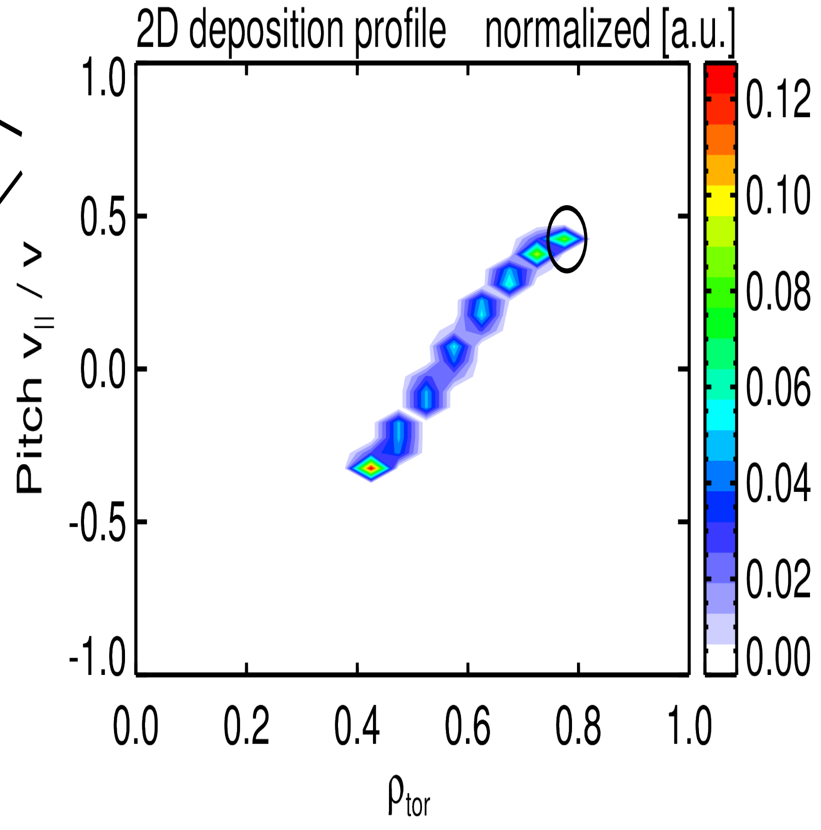
5. Applications

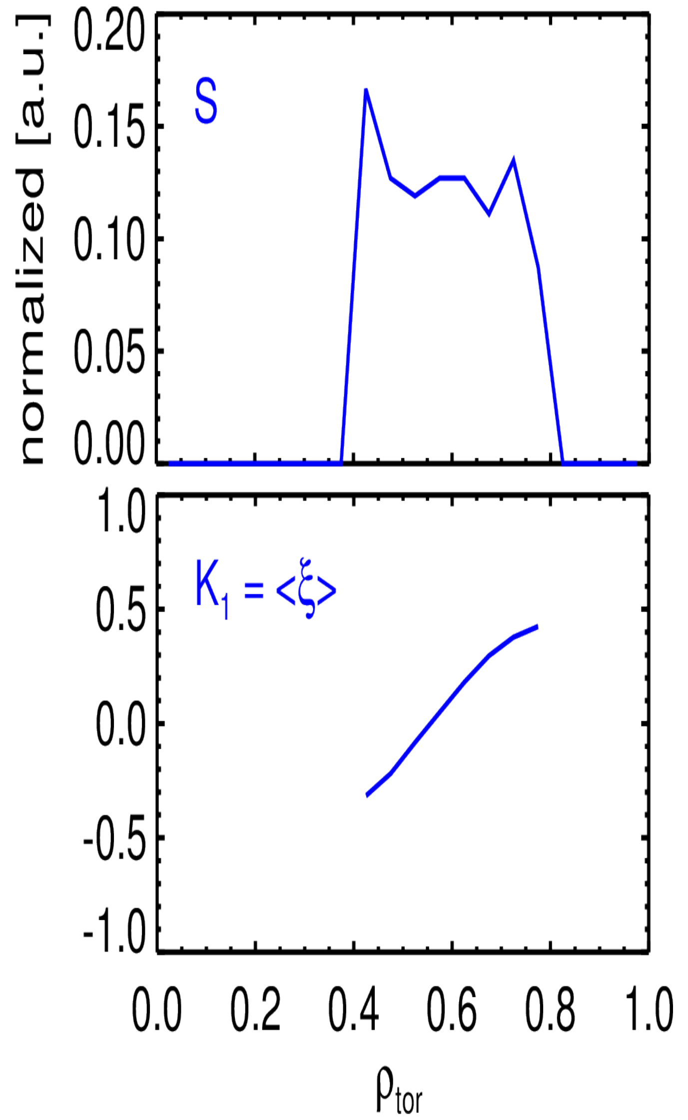
How to include the effect of first fast-ion orbit

- Orbit effects lead to a broadened deposition (towards the plasma center) and to changes of the pitch-distribution in the velocity space
- They can be taken into account, by averaging the deposition over the first orbit:
  - ➔ assume that slowing-down process starts on random position of first orbit
  - ➔ neglect orbit effects during slowing down



example for banana orbit:

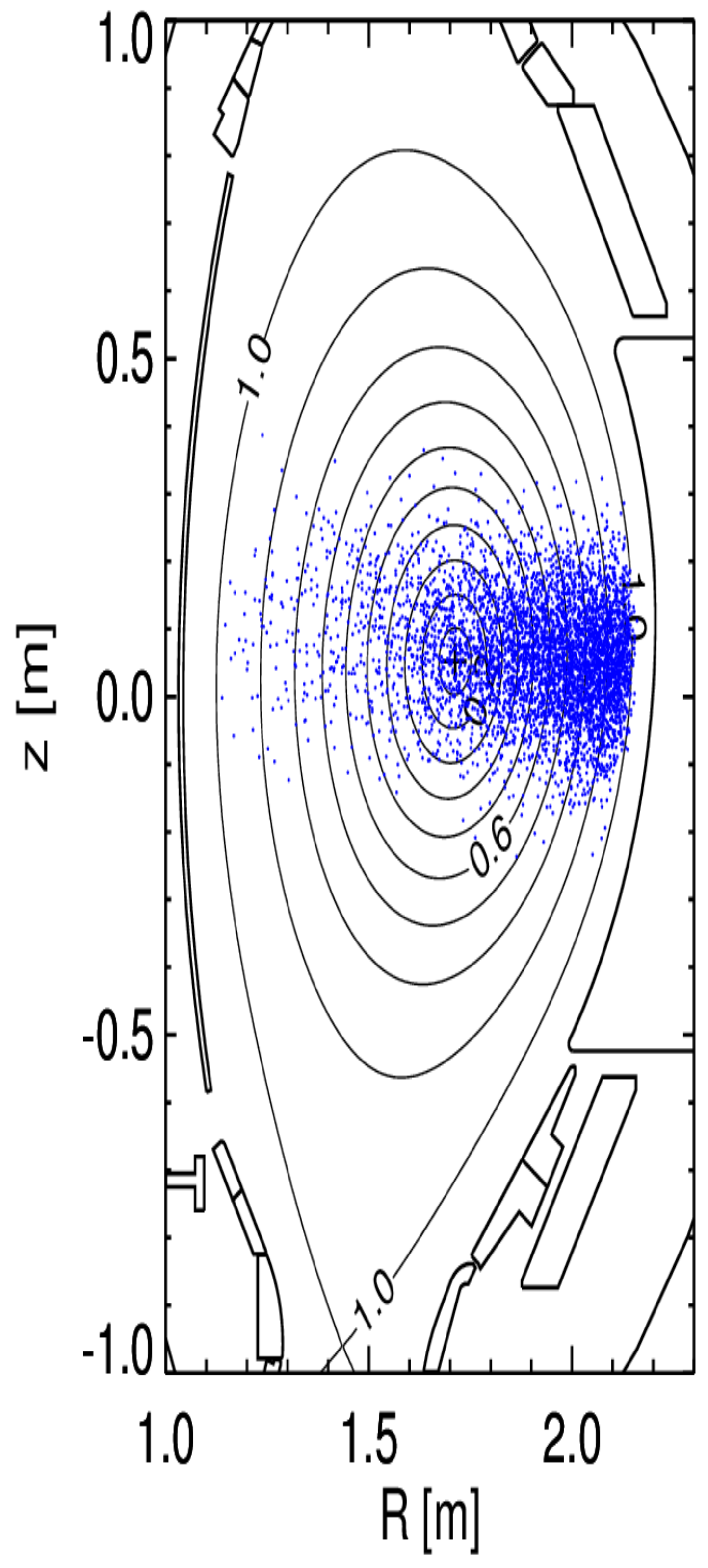




Monte-Carlo orbit-average is too slow for real-time applications

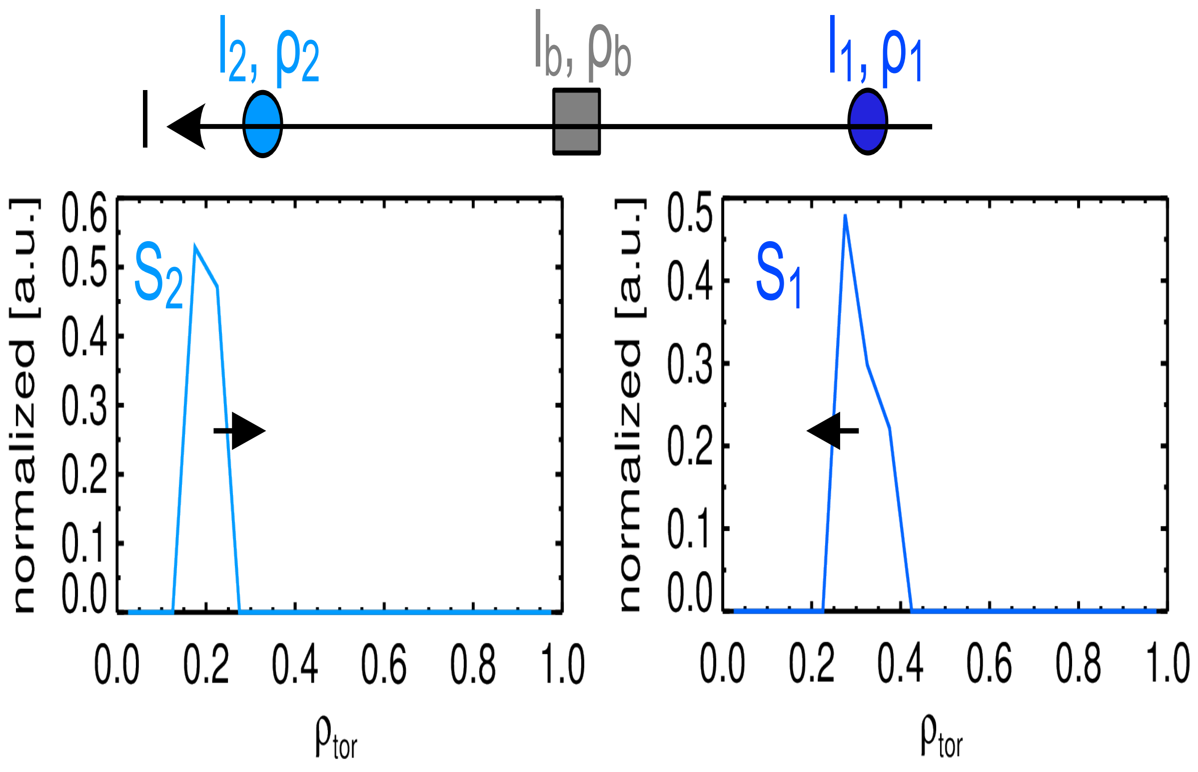
- Monte-Carlo orbit-average:
  - Take MC representation of birth distribution
  - Calculate orbit for each MC marker (e.g. ~5000)
  - → too slow for real-time purposes (takes ~1s)

- Possible solutions:
  - Either: Use approximation formulas for the orbits
  - Or: Reduce number of orbits (strongly)

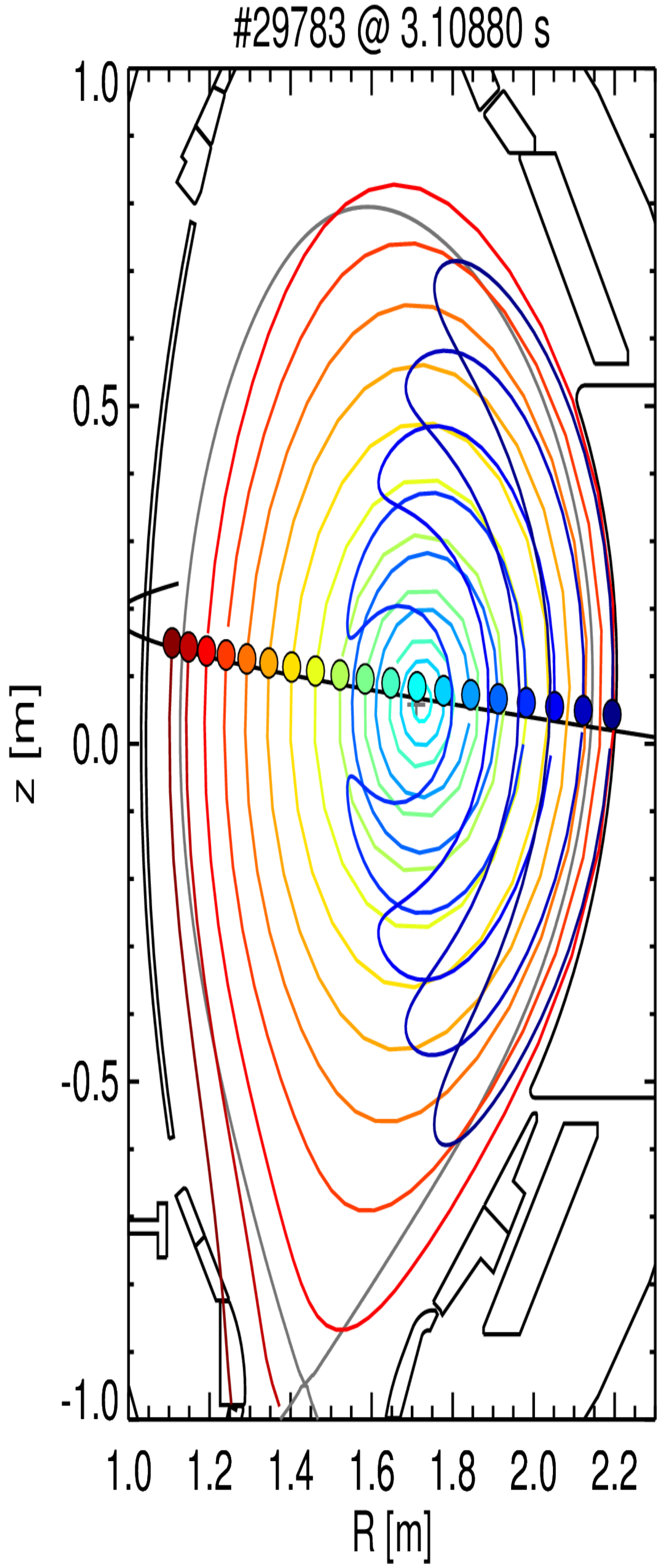


Orbit average in real-time

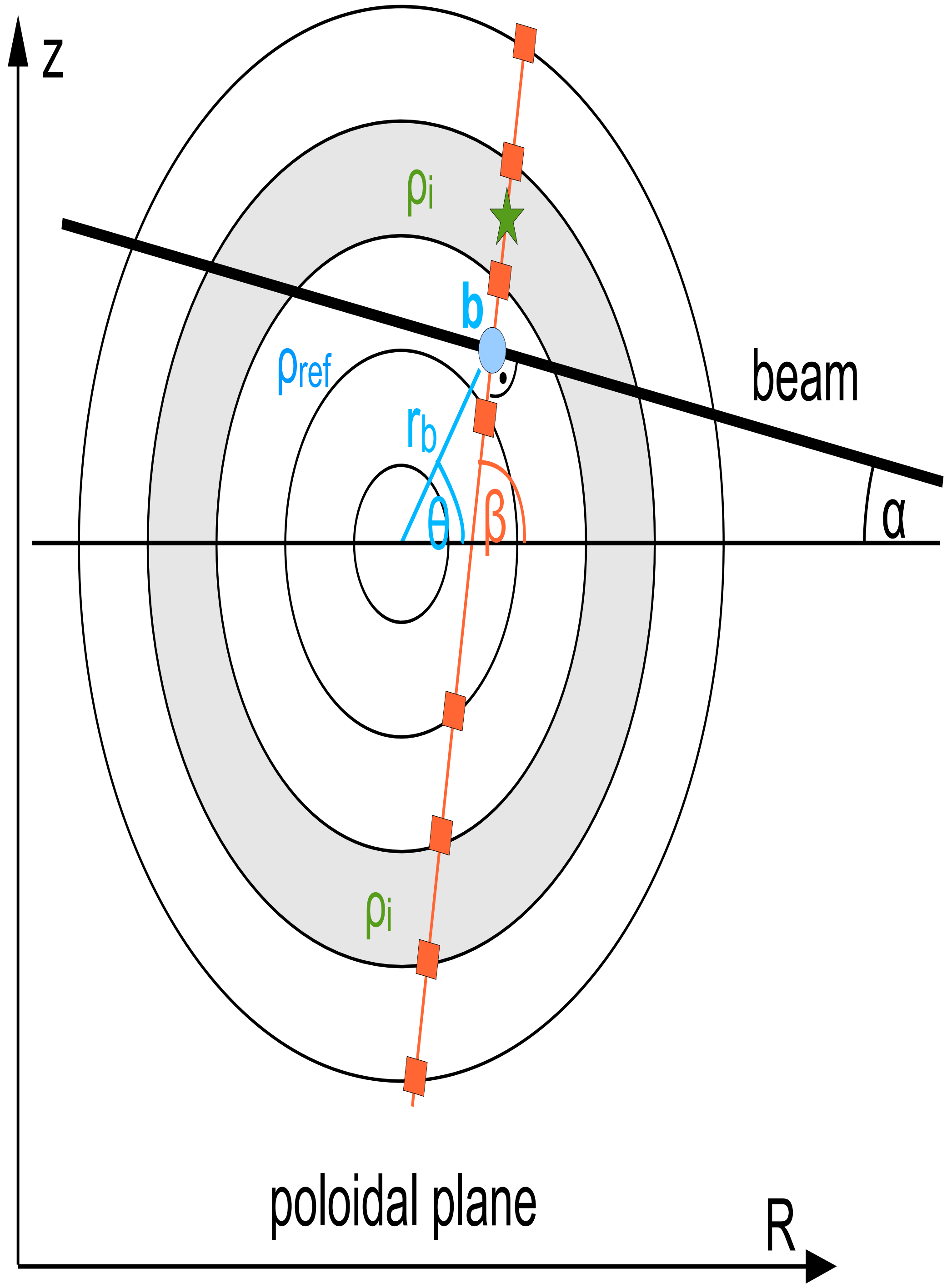
- Calculate orbit only every n-th grid point
  (4^{th} order Runge-Kutta guiding center integrator)
  Right: All calculated orbits for full energy component
- Here: 19 orbits x 3 energy components
  → possible within ~12 ms.
- In between: Shift neighboring profiles and interpolate linearly



$$S_b(\rho) = S_1(\rho - (\rho_b - \rho_1)) \cdot \frac{I_2 - I_b}{I_2 - I_1} + S_2(\rho - (\rho_b - \rho_2)) \cdot \frac{I_b - I_1}{I_2 - I_1}$$



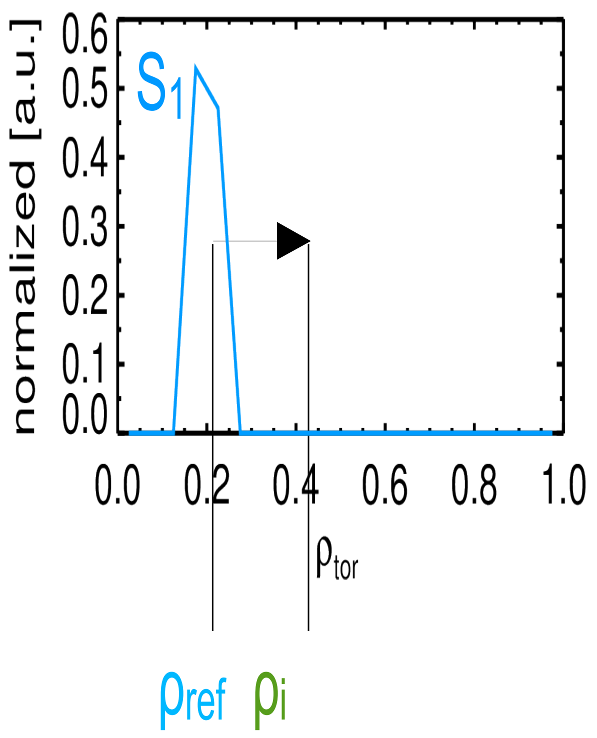
Orbit-average: compatible with beam-width model



- Up to now, we have calculated the orbit-average along the beam (at **b**).
- For the beam-width correction, we need to extrapolate from the ρ-cell containing **b** (ρ_{ref}) along the orange line to other radial cells
- E.g. from ρ_{ref} to ★(ρ_i):

$$S_i(\rho) = S_b(\rho - (\rho_i - \rho_{ref}))$$

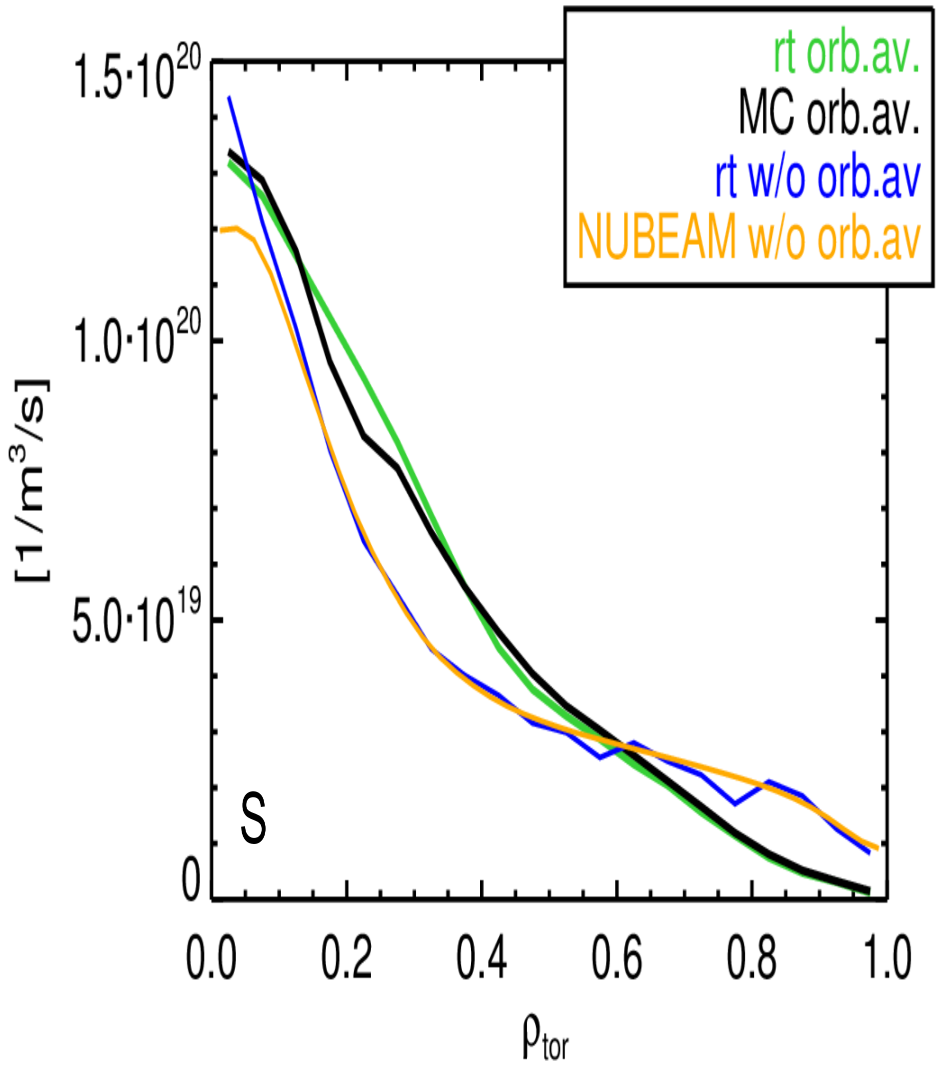
(similar to the interpolation method)



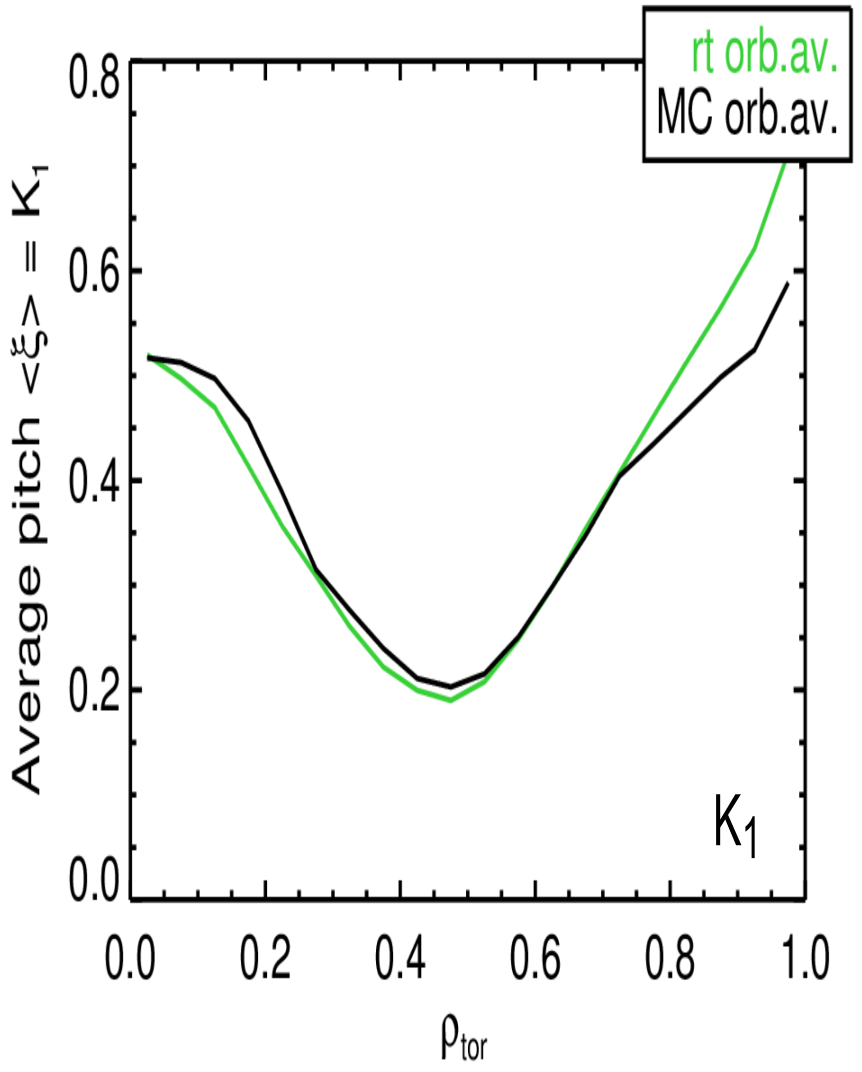
Results of „RABBIT orbit average“ in good agreement with MC orbit average

- Test accuracy of the RABBIT rt orbit average:
  Compare it to Monte-Carlo orbit average (including fully realistic NBI geometry)
- Very good agreement is found, despite orders of magnitude difference in calculation time (~5000 orbits vs. ~60 orbits)

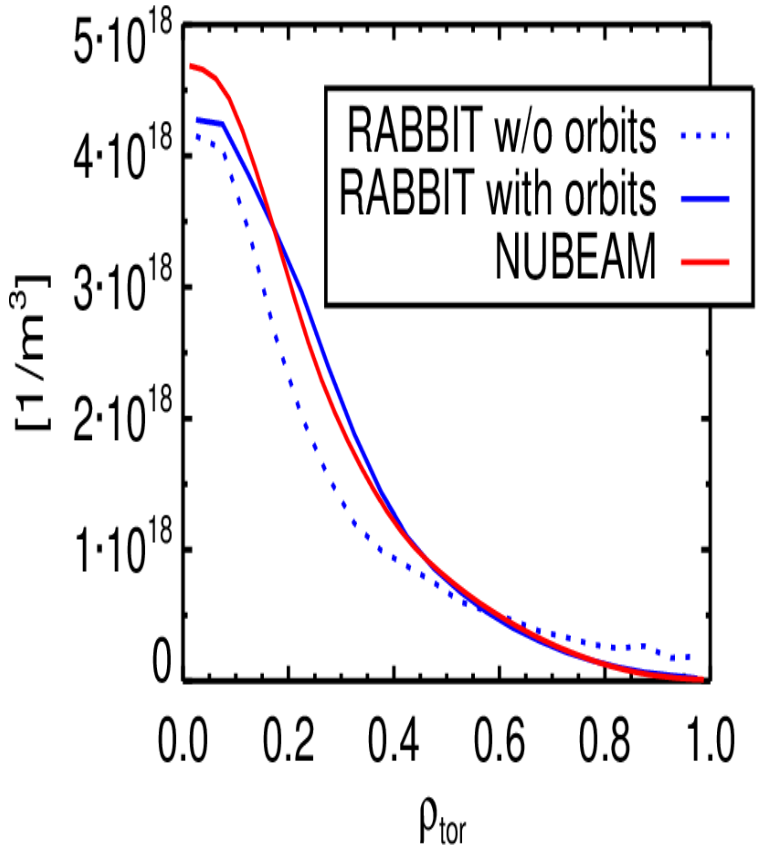
Birth profile (sum over all E components):



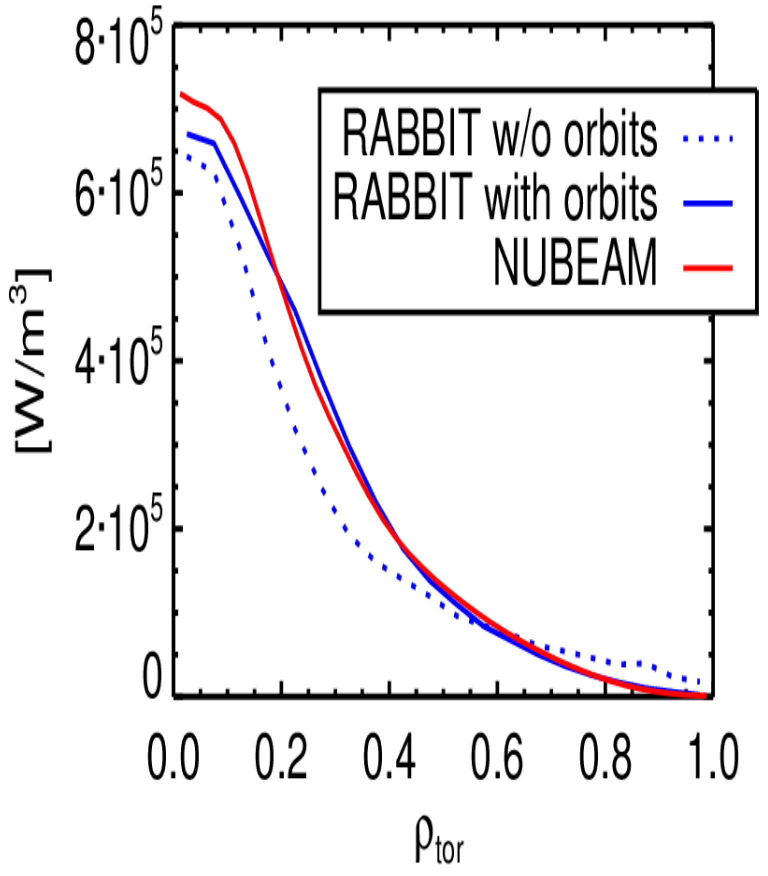
Average pitch $v_{||}/v$ of full-E component:



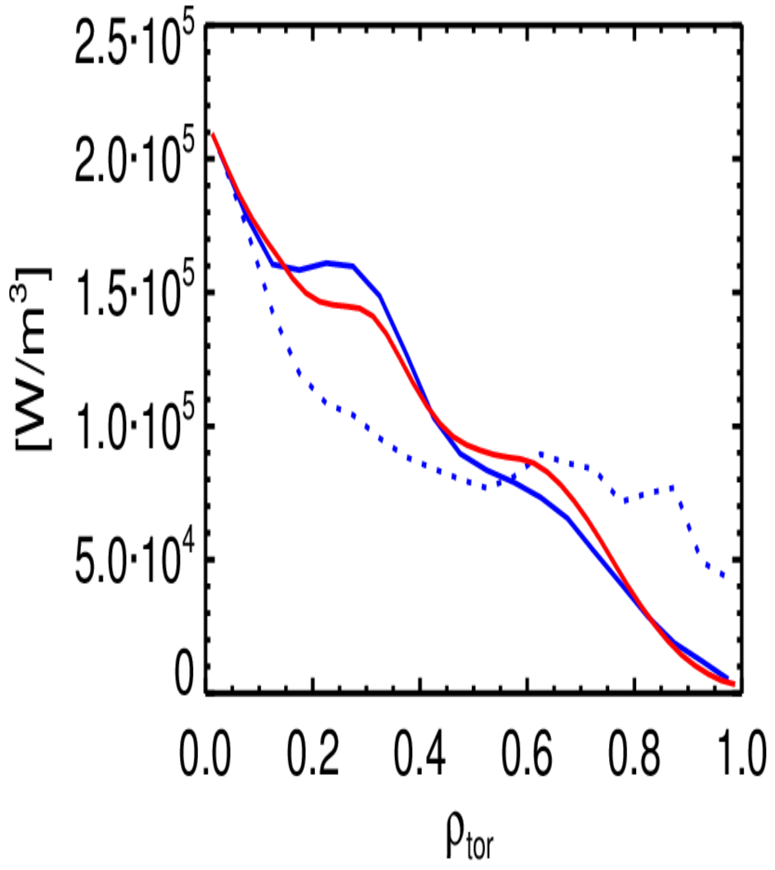
Comparison to NUBEAM



Fast-ion density

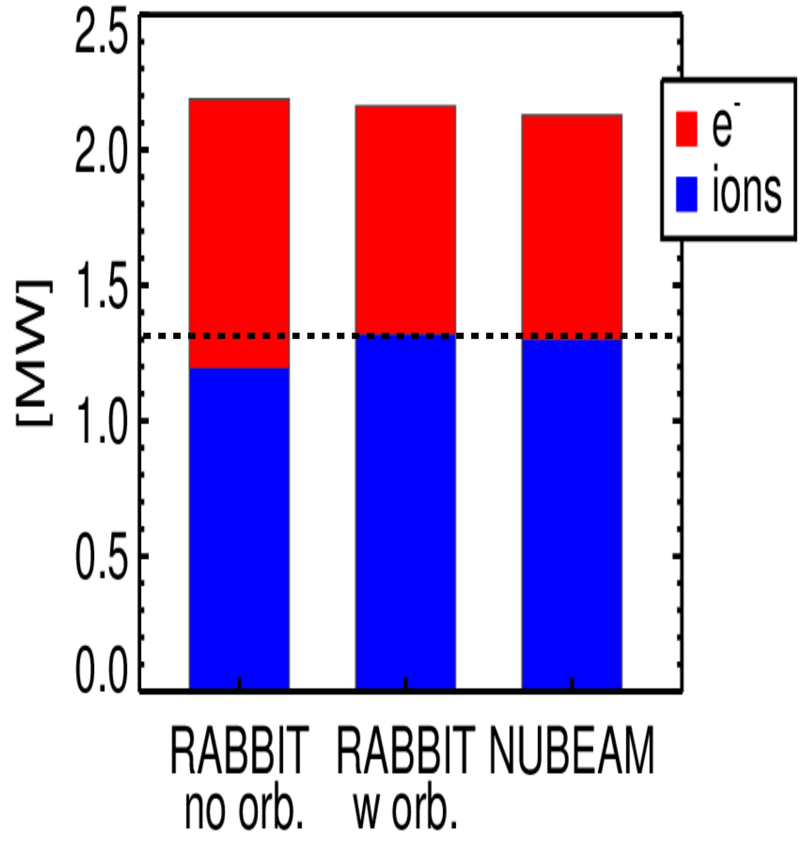


Heating to ions



Heating to electrons

- Orbit-average leads to good agreement in profile shape
- Slight deviations remain in plasma center, affecting only small fraction of plasma volume
- Orbit-average has also an impact on volume-integrated heating distribution to electrons/ions and improves agreement



Kinetic equation - outline

Kinetic equation for distribution function f(x, v, t)

$$\frac{\partial f}{\partial t} + \underbrace{\boldsymbol{v}\cdot\nabla_{\boldsymbol{x}} f + \boldsymbol{a}\cdot\nabla_{\boldsymbol{v}} f}_{} = \nabla_{\boldsymbol{v}}\cdot\Gamma_c(f) \quad + \text{Source}$$

3. Orbit effects

$$\boldsymbol{a} = \frac{q}{m}\left(E + \frac{\boldsymbol{v}}{c}\times B\right)$$

4. Time dependence
(=0 for steady state
solution)

2. collisions
(e.g. slowing down,
pitch angle scattering)

1. Source = NBI
deposition

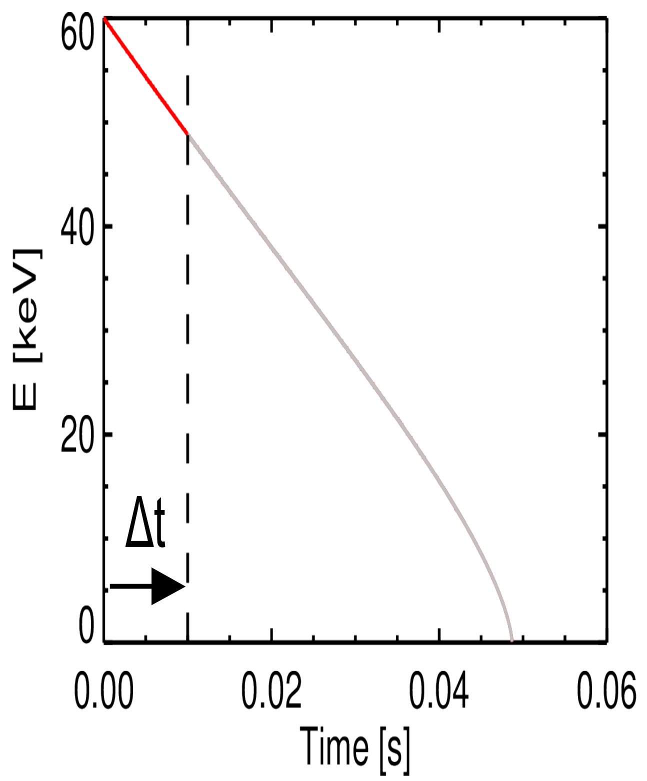
5. Applications

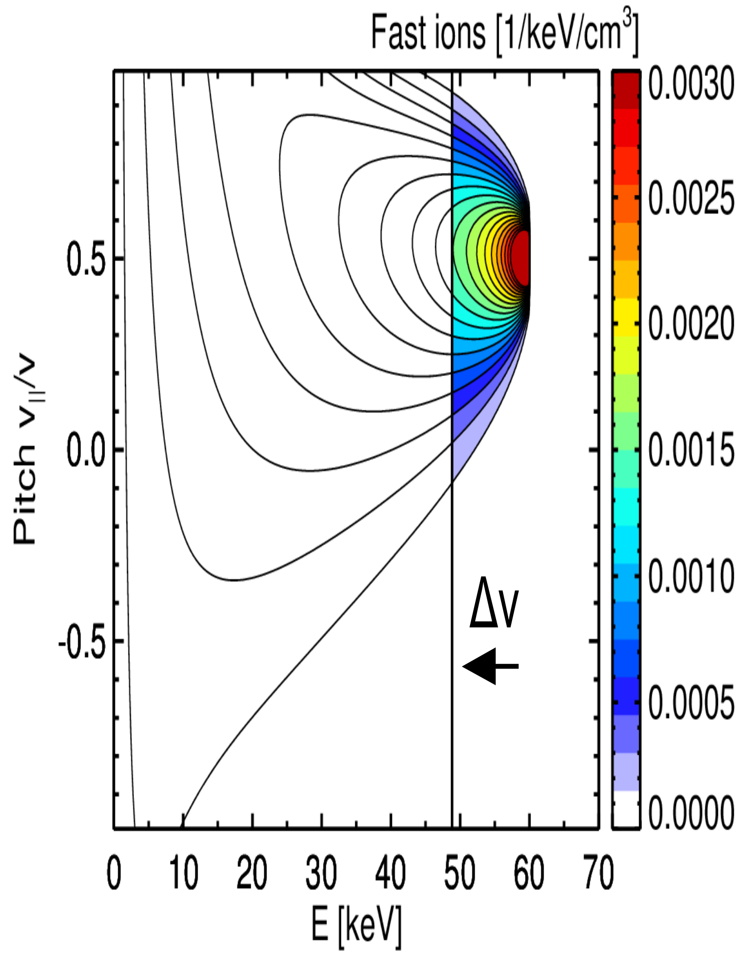
Time dependence

- For time-dependent simulation: Discrete time steps Δt
- Calculate how far the fast-ions slow down during time-step:

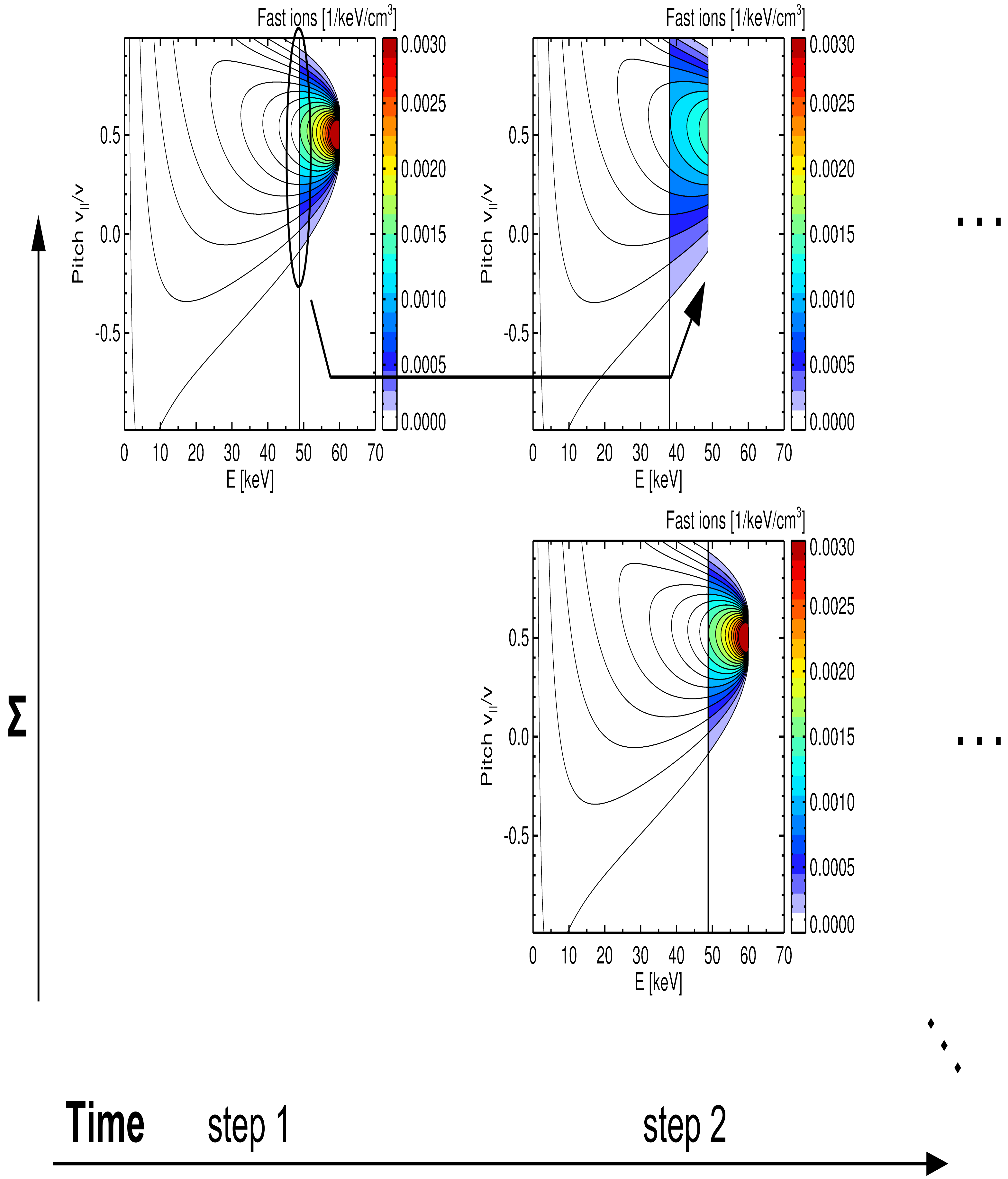
$$v_{\text{final}}^3 = \left(v_{\text{start}}^3 + v_c^3\right) \cdot \exp\frac{-3 \cdot \Delta t}{\tau_s} - v_c^3$$

→ multiply steady state solution with box function





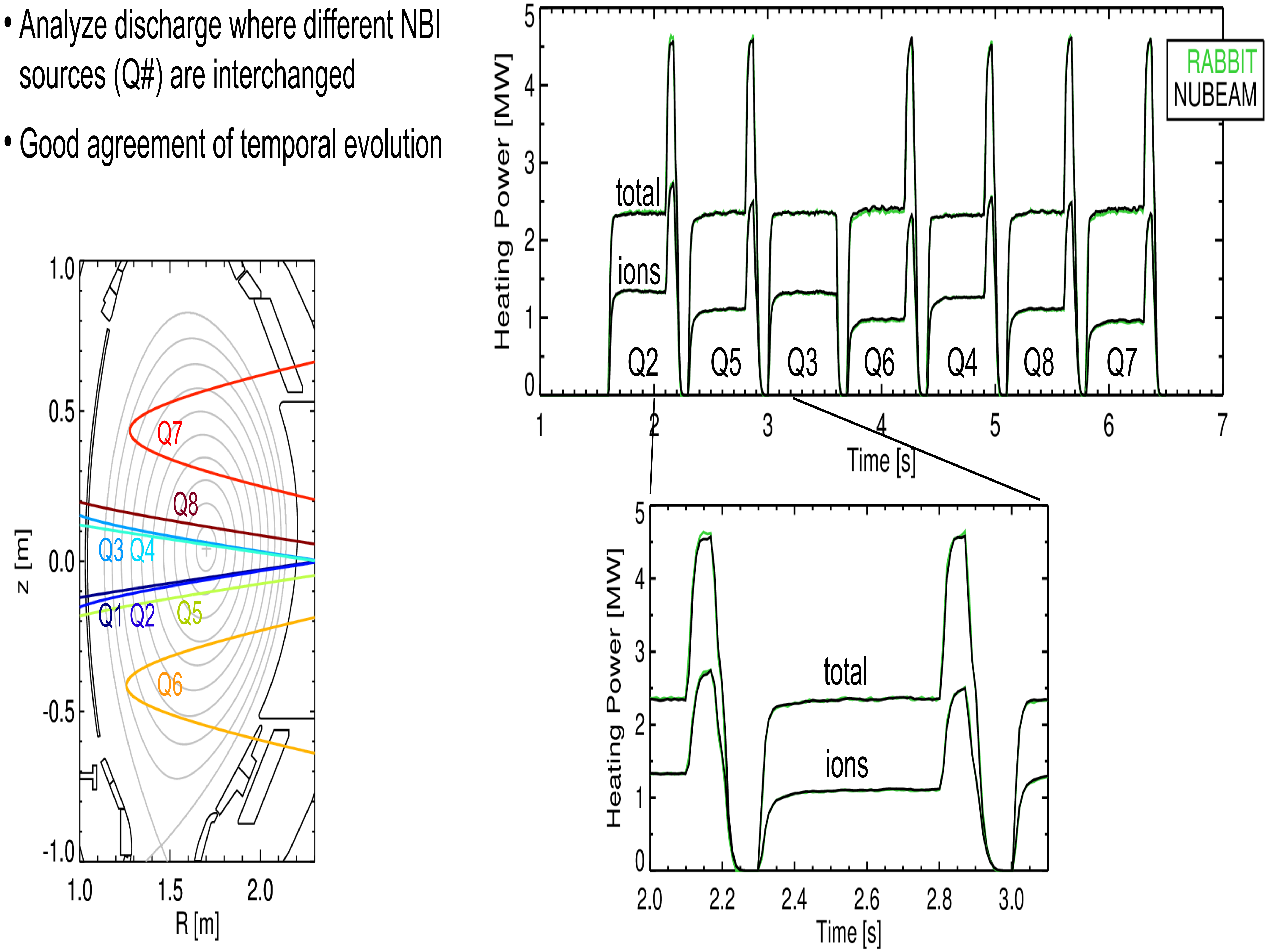
Time dependence via train of fast-ion pulses



- Final state of „step 1“ is starting point of „step 2“
- If beam is still turned on in „step 2“, add a new pulse at nominal injection energy
- continue ...
  (add new rows each time-step, sum over rows)

Comparison of time evolution with NUBEAM

- Analyze discharge where different NBI sources (Q#) are interchanged
- Good agreement of temporal evolution



Kinetic equation - outline

Kinetic equation for distribution function f(x, v, t)

$$\frac{\partial f}{\partial t} + \underbrace{\boldsymbol{v}\cdot\nabla_{\boldsymbol{x}} f + \boldsymbol{a}\cdot\nabla_{\boldsymbol{v}} f}_{} = \nabla_{\boldsymbol{v}}\cdot\Gamma_c(f) \quad + \text{Source}$$

3. Orbit effects

$$\boldsymbol{a} = \frac{q}{m}\left(\boldsymbol{E} + \frac{\boldsymbol{v}}{c}\times\boldsymbol{B}\right)$$

4. Time dependence (=0 for steady state solution)

2. collisions (e.g. slowing down, pitch angle scattering)

1. Source = NBI deposition

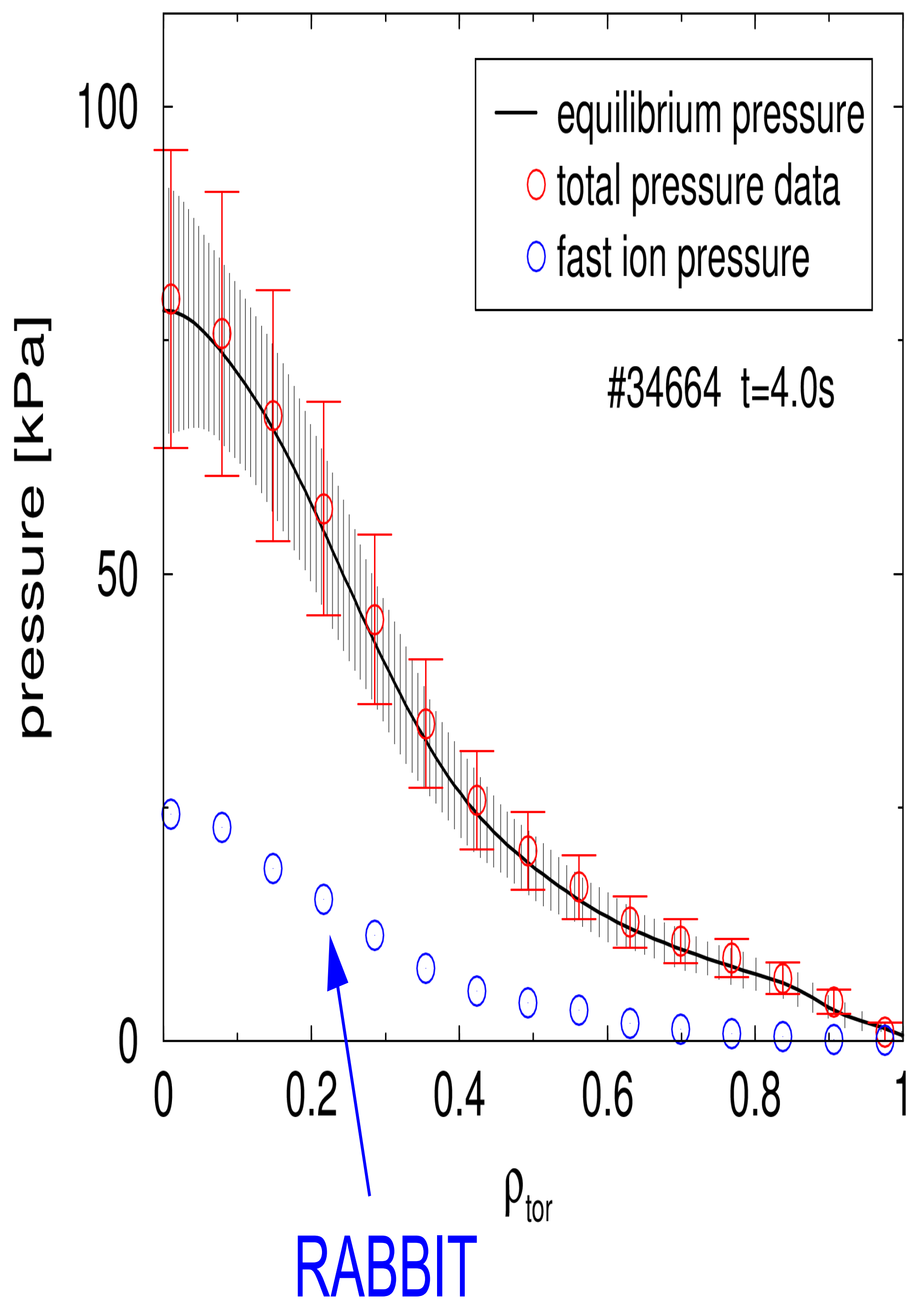
5. Applications

RABBIT for equilibrium reconstructions with the IDE code

Equilibrium reconstruction: Solve Grad-Shafranov-Schlüter equation for ψ:

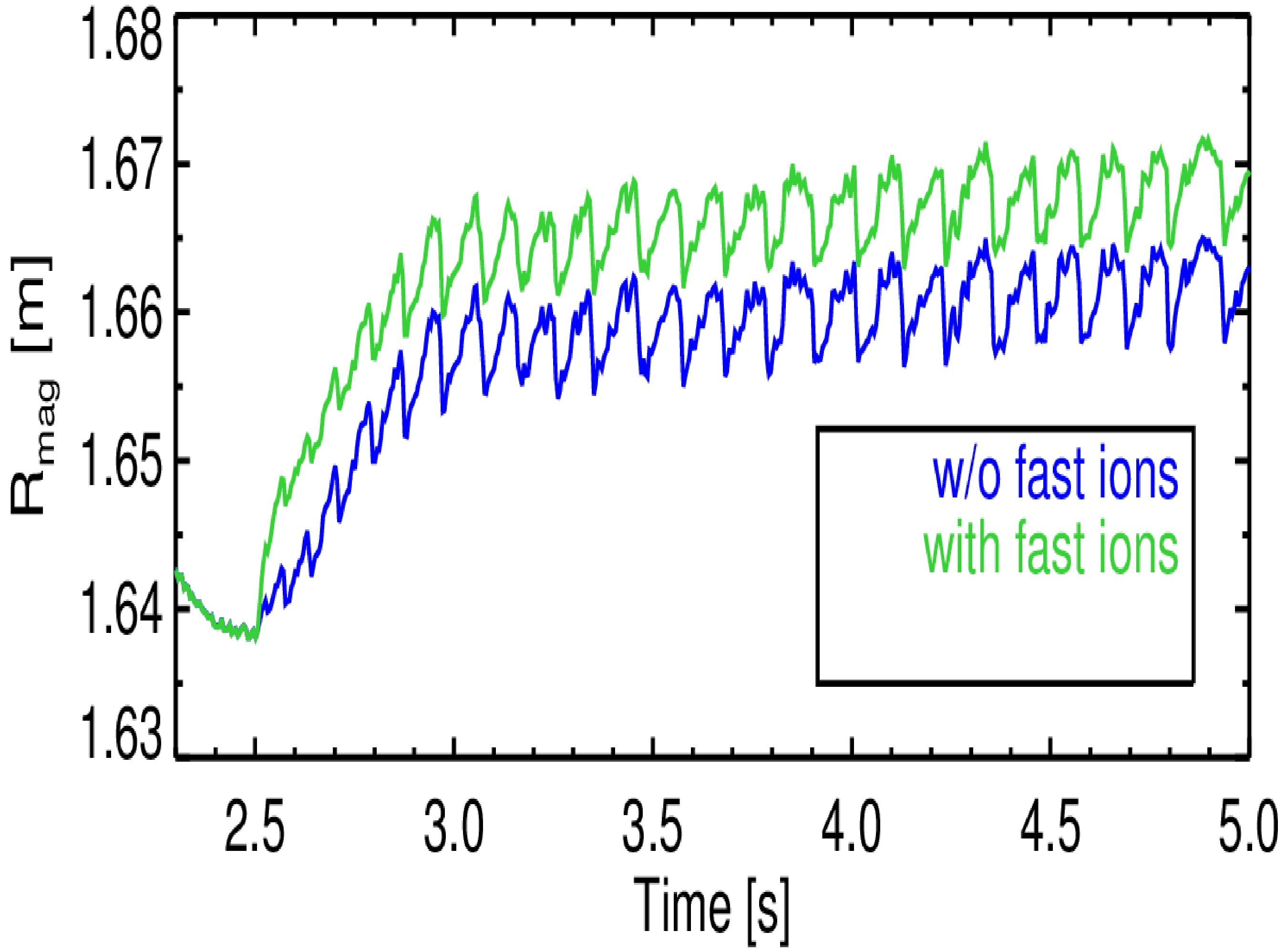
$$\Delta^* \Psi = R^2 \mu_0 \frac{\mathrm{d}p}{\mathrm{d}\Psi} + F \frac{\mathrm{d}F}{\mathrm{d}\Psi}$$

- IDE code: Equilibrium reconstruction based on integrated data analysis [R. Fischer, 2016, FST]
- Fast-ions from NBI can contribute significantly to the **pressure** p(ψ)
- Neutral beam **current** drive is relevant for current diffusion equation
- With RABBIT these profiles can be calculated routinely and also directly after the discharge



RABBIT for equilibrium reconstructions with the IDE code

- Fast-ion pressure
  → increases Grad-Shafranov shift
  → correction for magnetic axis
- Relevant for correct interpretation of diagnostics (e.g. MSE)
- In this case: Sawtoothing plasma, sawtooth-induced fast-ion redistribution is relevant, too

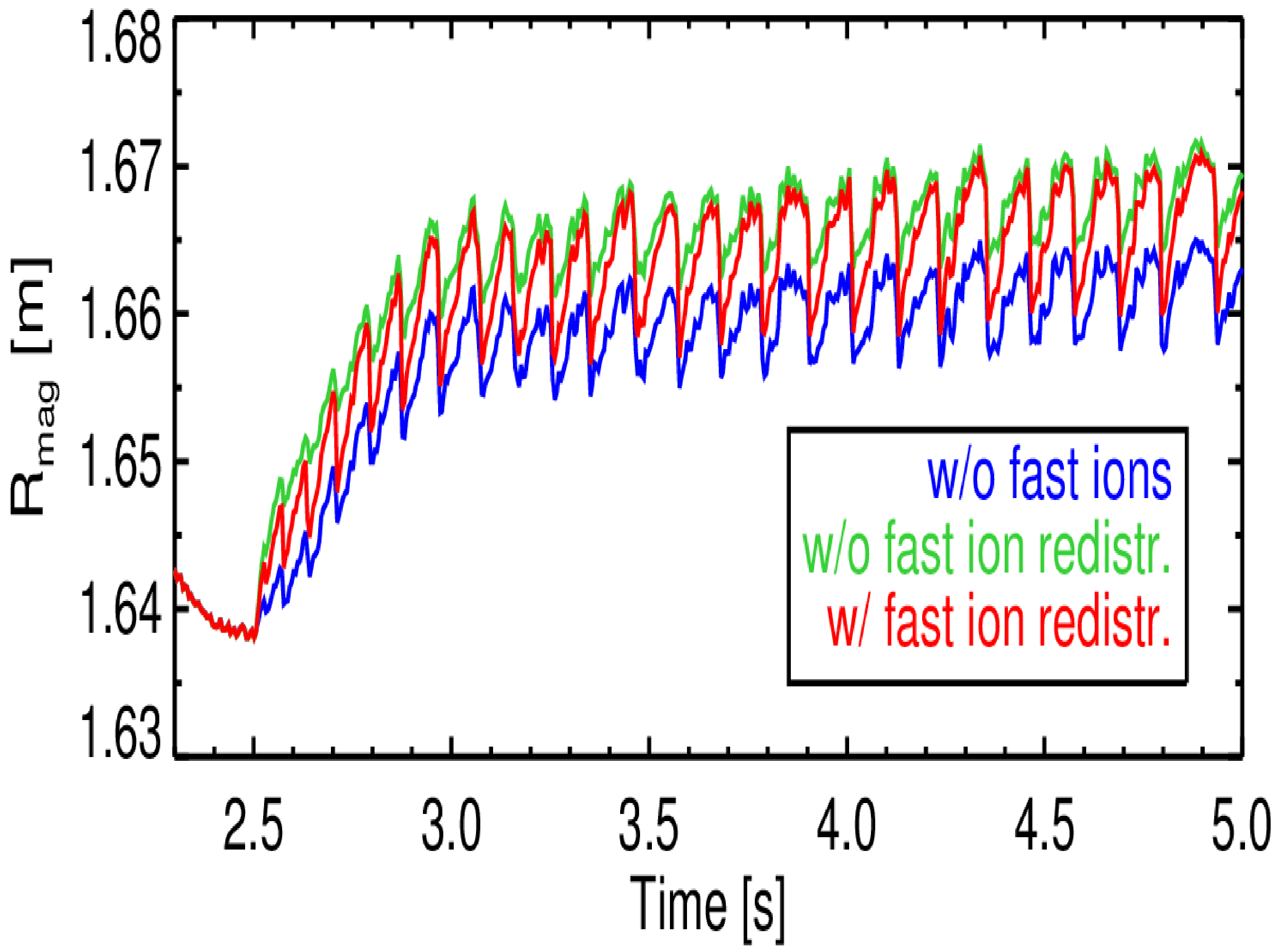


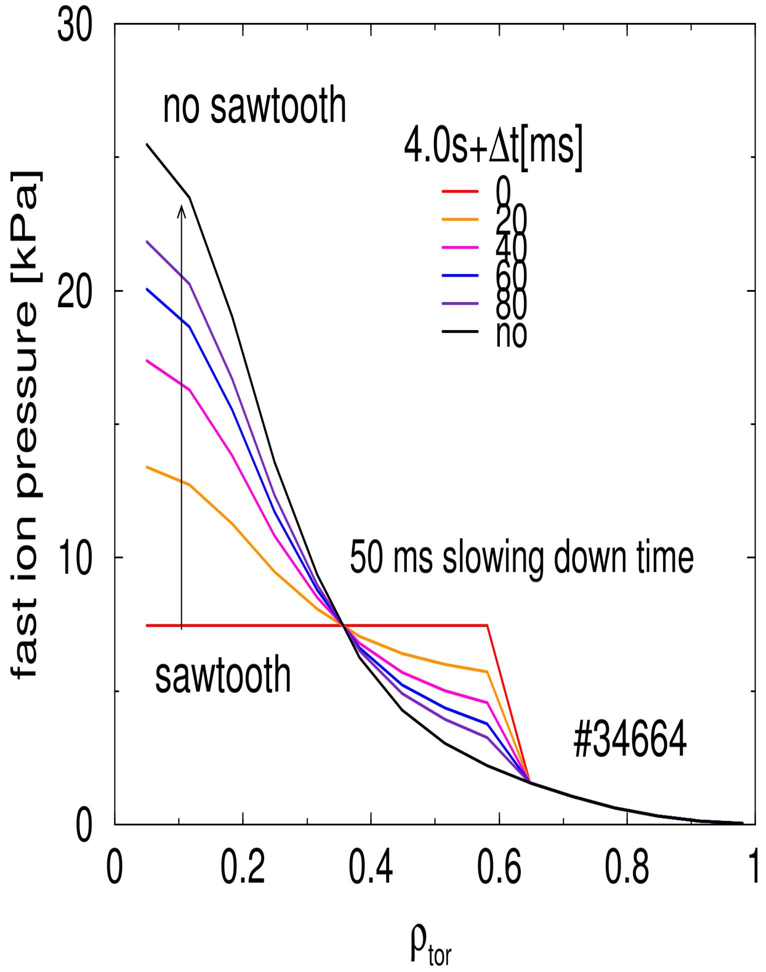
RABBIT for equilibrium reconstructions with the IDE code

- Fast-ion pressure
  → increases Grad-Shafranov shift
  → correction for magnetic axis
- Relevant for correct interpretation of diagnostics (e.g. MSE)
- In this case: Sawtoothing plasma, sawtooth-induced fast-ion redistribution is relevant, too c

  → here modeled by post-processing RABBIT profiles

  → Outlook: implement redistribution directly in RABBIT.

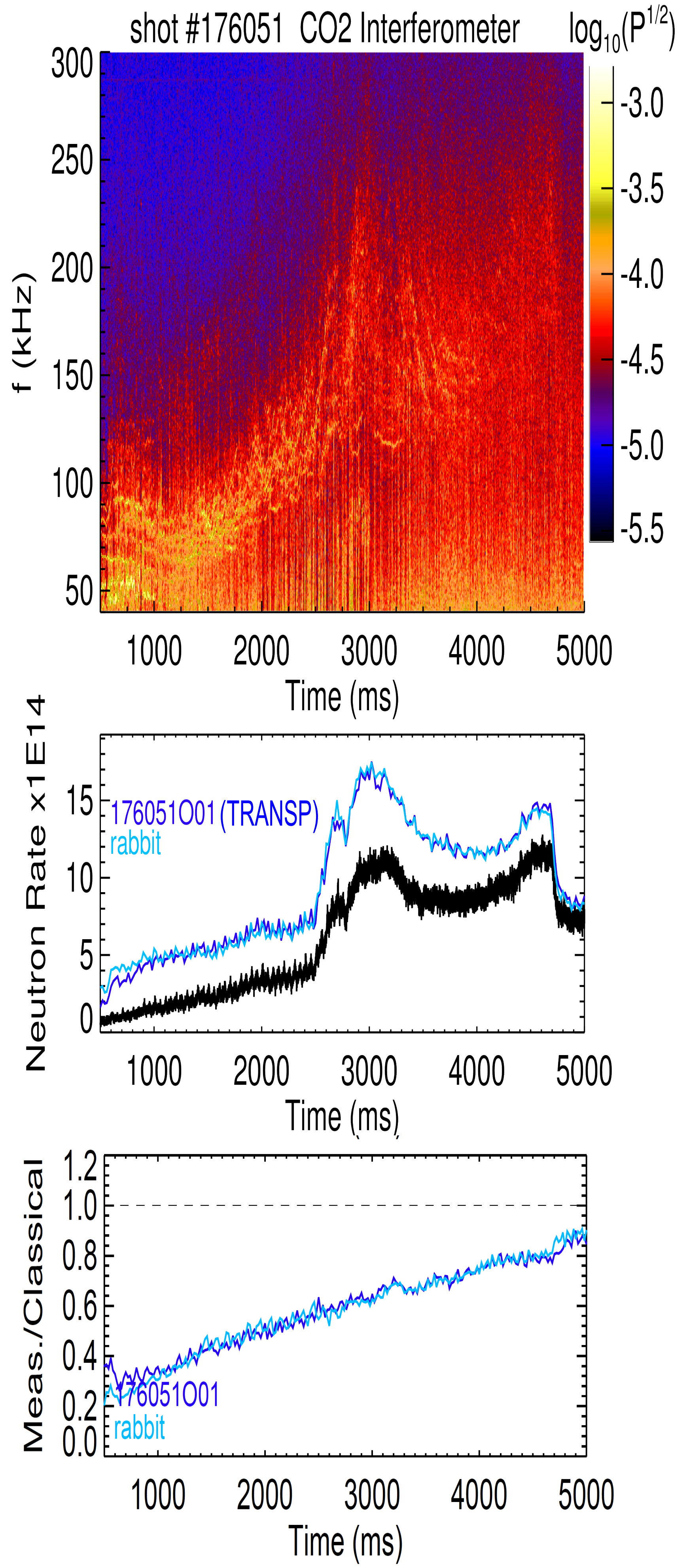




Detect AE-induced fast-ion transport by analyzing neutron rates

- Alfvén eigenmodes (AE) can be measured directly by fluctuation diagnostics
- Do they cause strong fast-ion transport?
  → can be assessed by comparing measured neutron rate with neo-classical prediction
- With RABBIT, this information is available directly after the discharge or even in real-time

- Used in scenario development for reversed shear steady state scenario experiments
  → Goal: Reduce fast-ion transport to optimize perfomance
  → RABBIT useful for decision making between discharges



Database comparison between TRANSP and RABBIT

- Comparison between RABBIT and TRANSP for all shots from the experimental session
  → Good agreement

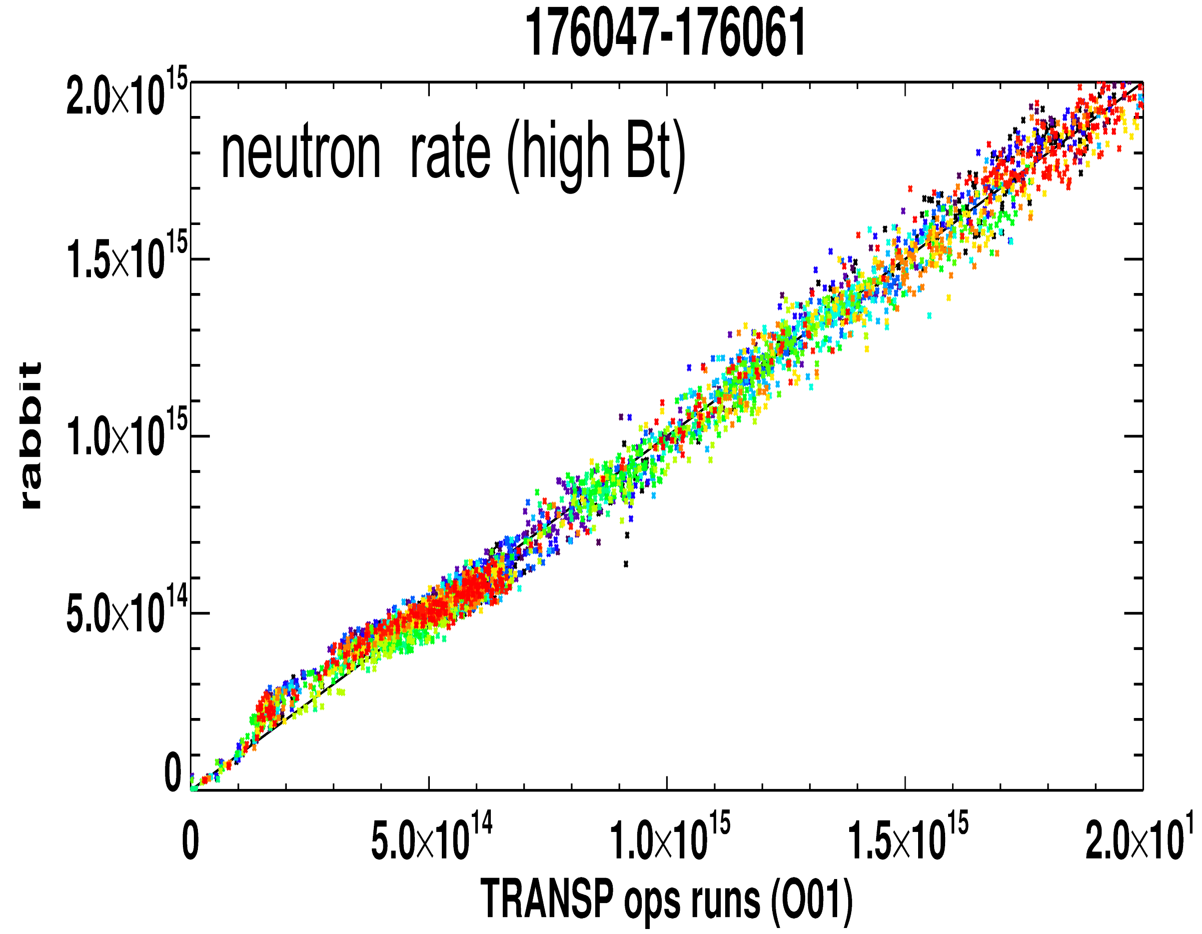
- Outlook: Use for real-time control of AE modes

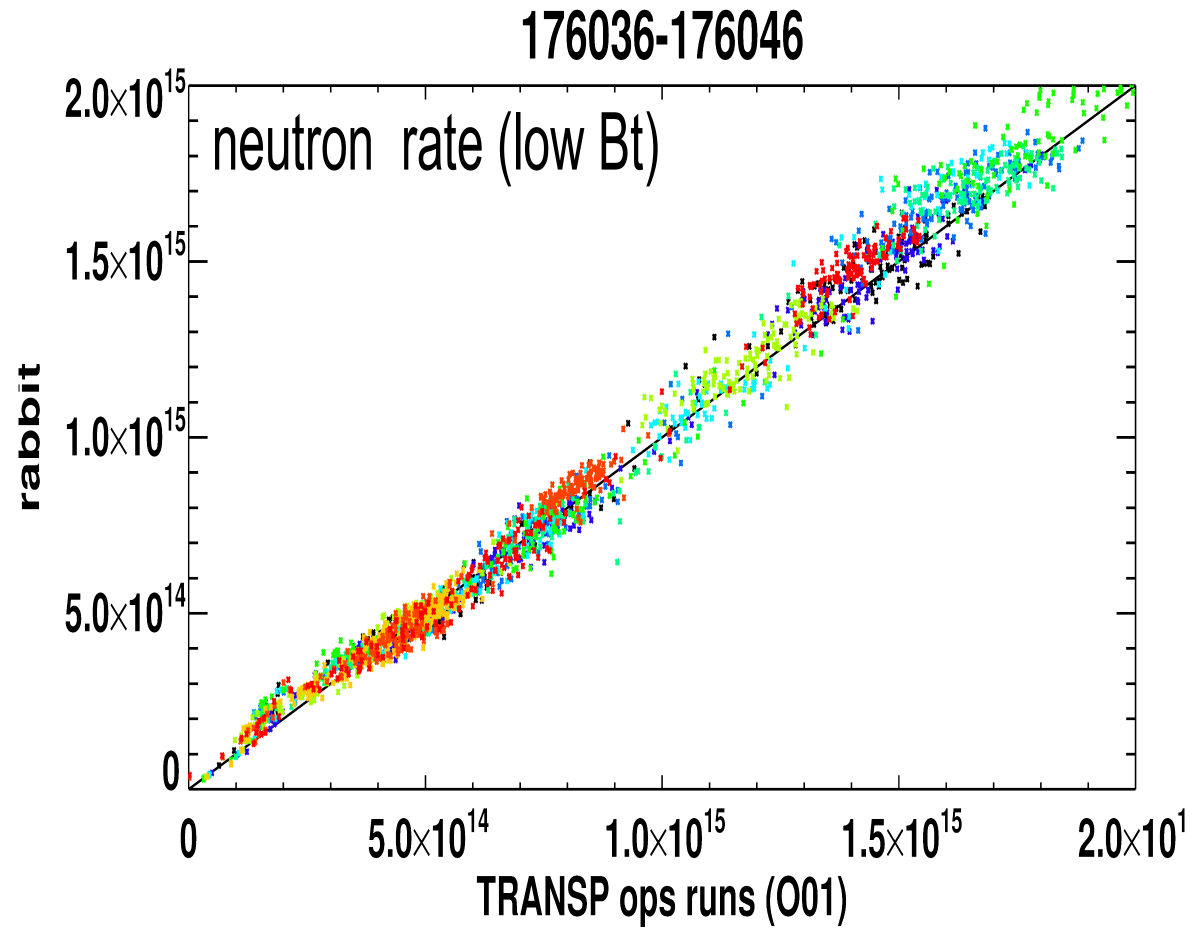
  → use ECE to detect AE amplitude

  → use RABBIT to detect detrimental fast-ion transport

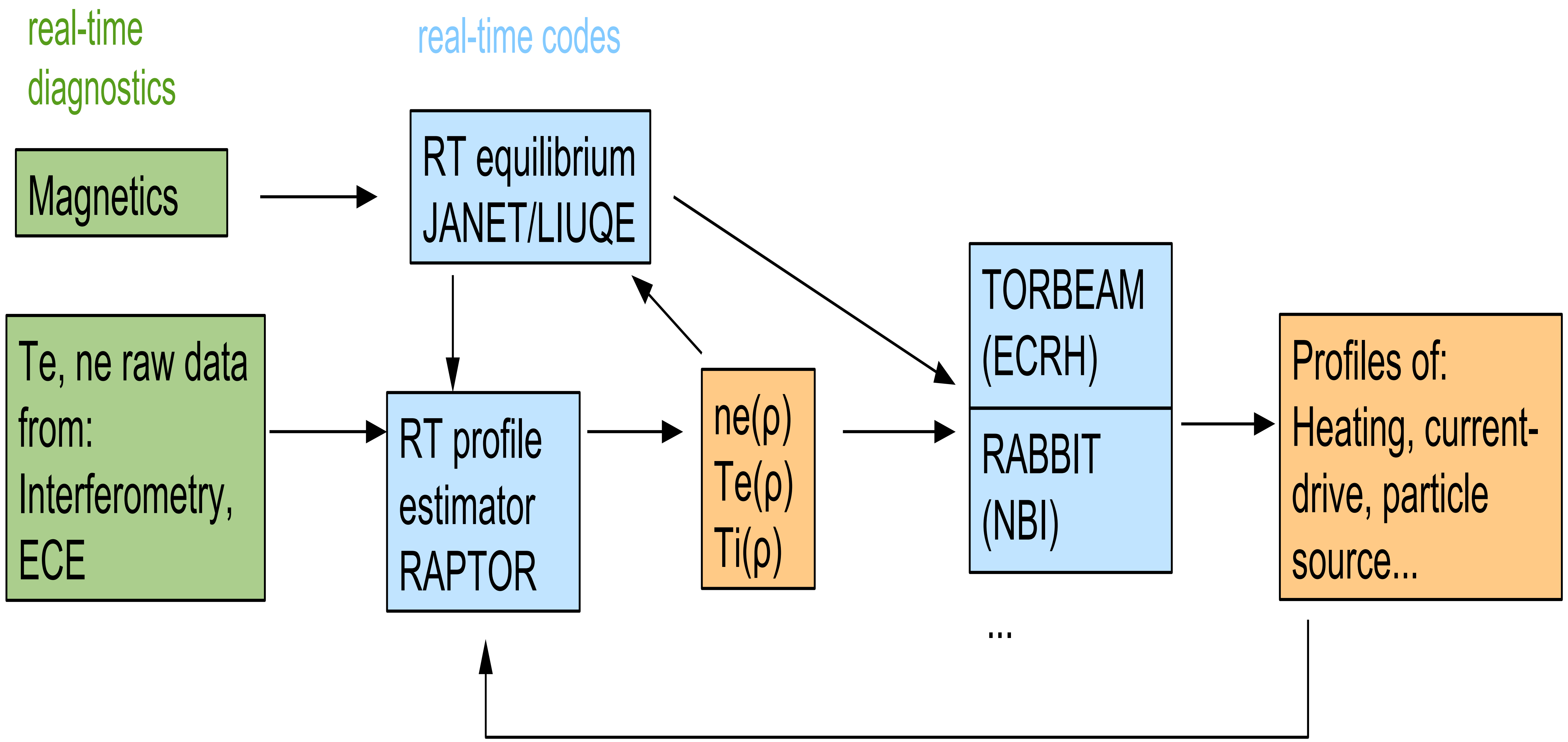
  → trigger counter-measures, e.g. reduce NBI power (=AE drive)

- Use for data-mining (plan to run Rabbit on every historic DIII-D discharge)





Real-time implementation of RABBIT on ASDEX Upgrade and TCV



- RAPTOR is 1D real-time transport solver → relies on realistic inputs of sources by auxiliary heating
- Status of RABBIT implementation:
  Hardware is installed, software is being finalized, first tests/results are expected in few weeks

Summary – RABBIT: A high-fidelity real-time NBI code

Three parts:	Calc. time:	Method:
• 1. Beam attenuation – birth profile	~ 7 ms	Simplified geometry (“thin beam”), analytic treatment of beam width
• 2. Orbit average	~ 12 ms	Conventional GC-integrator, but for very few orbits (~20 per E-comp.)
• 3. Time dep. solution of FP equation	~ 1 ms	Fully analytic

- In total ~20 ms per time-step (1 thread per beam) - faster than NUBEAM by roughly a factor of ~1000
- Good agreement with NUBEAM e.g. for heating profiles, neutron rates, fast-ion pressure
  tested so far on ASDEX Upgrade, DIII-D and JET

Applications / Outlook:

- For improved equilibrium reconstruction (e.g. IDE)
- Neutron rates allow to assess fast-ion transport (e.g. at DIII-D, intershot and real-time)
- Real-time control applications (with RAPTOR) at ASDEX Upgrade and TCV
- In ASTRA for the currently developed ASDEX Upgrade flight simulator
- In integrated modeling frameworks (IMAS, ETS, JINTRAC, OMFIT, …)

Backup

CPU hardware

Intel Xeon E5-2680 v3 (2.5GHz) CPUs,