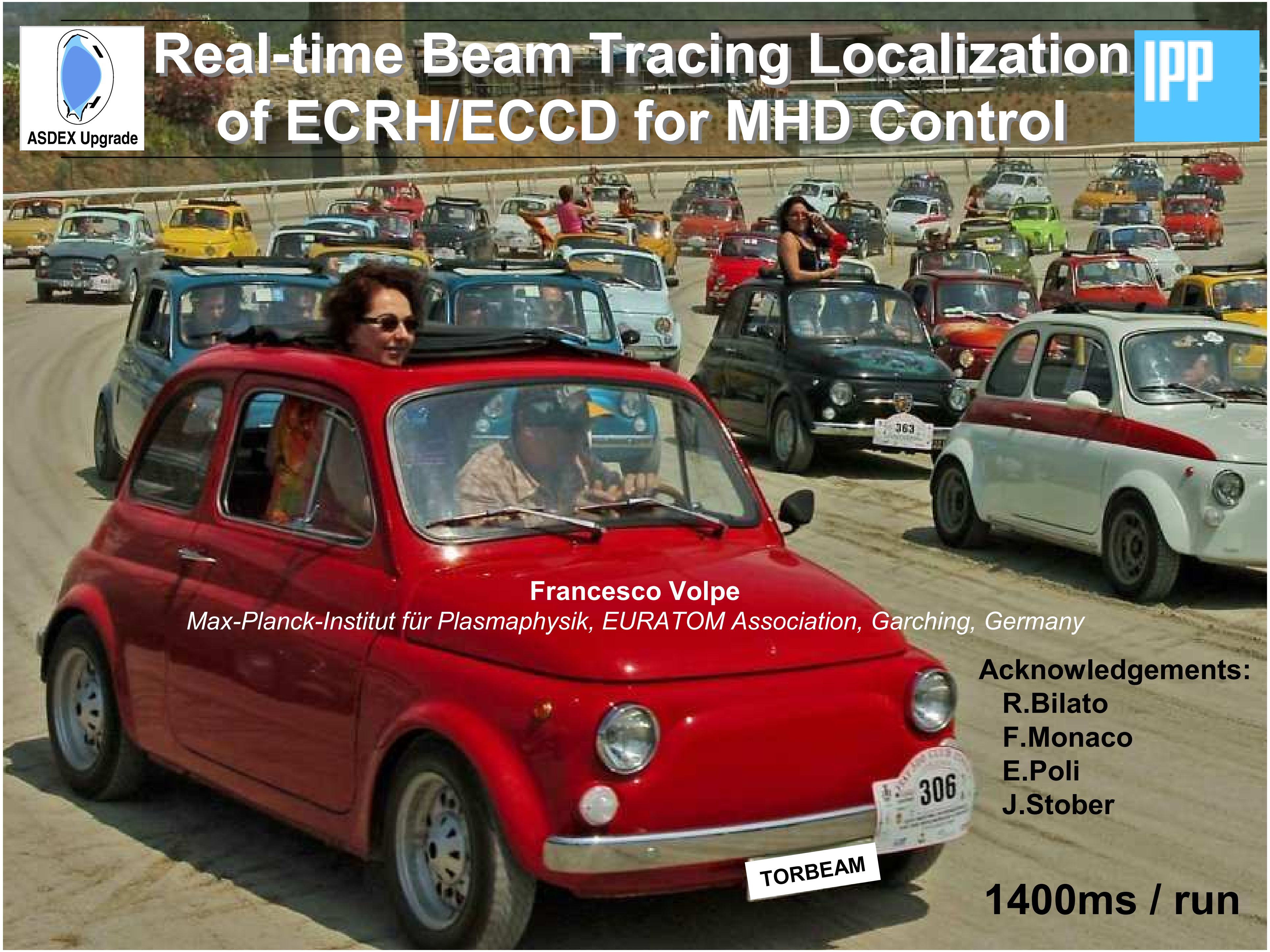
Real-time Beam Tracing Localization of ECRH/ECCD for MHD Control



Francesco Volpe
Max-Planck-Institut für Plasmaphysik, EURATOM Association, Garching, Germany

Acknowledgements:
R.Bilato
F.Monaco
E.Poli
J.Stober

1400ms / run

ASDEX Upgrade

Outline

- Real-time Steering: Motivation and Limits
- Experimental Setup
- Timescale: physical requirements and technological constraints
- Look-Up Table
- Real-time Ray-tracing (Rt^2)
- Conclusions and Future Work

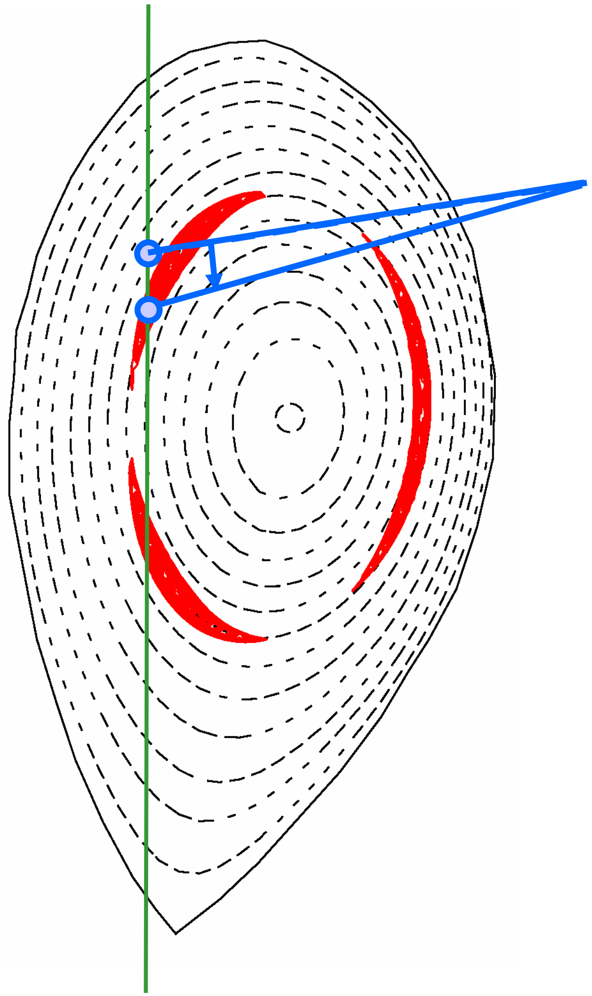
Real-time Steering is the only viable technique for aligning ECCD to NTMs in ITER

Two actuators proved so far:

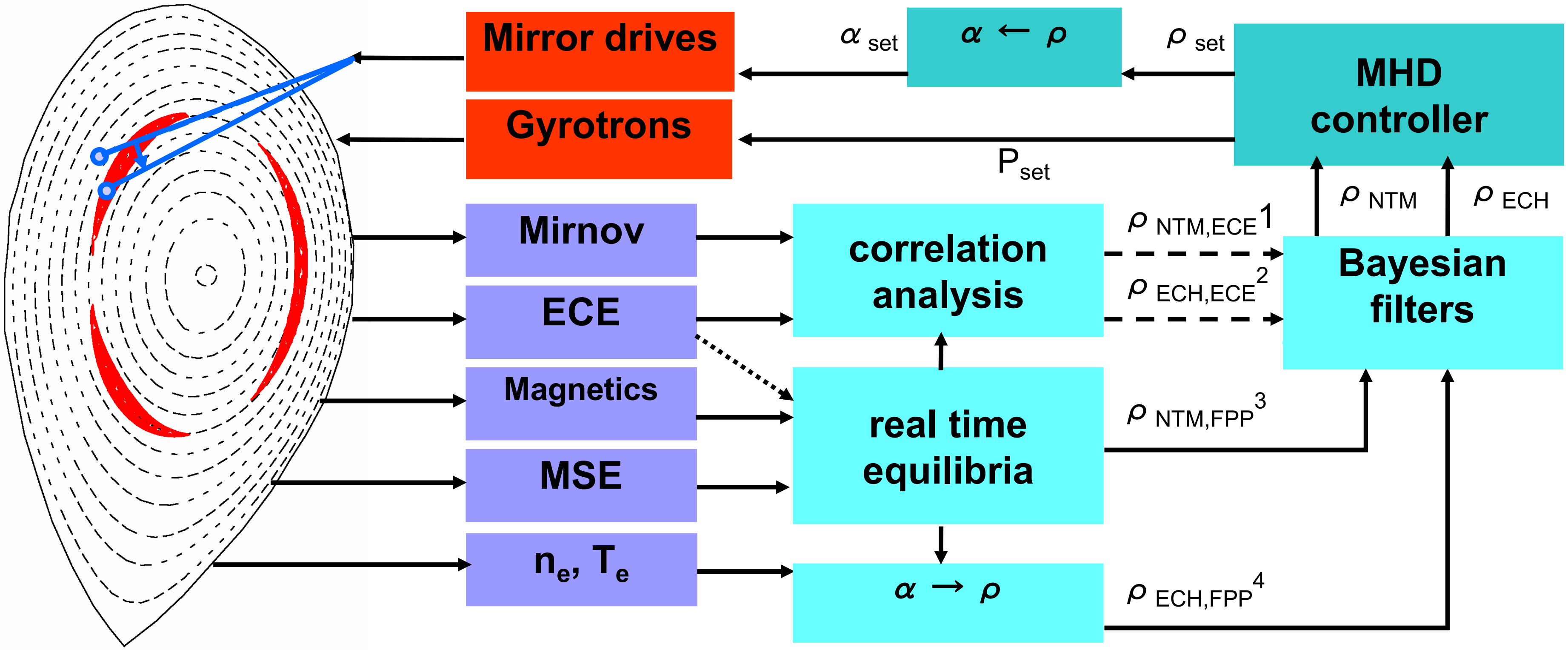
1. Varying B_T to move EC resonance (AUG & DIII-D)
   - *B_T fixed –within shot- in superconducting devices like ITER*
2. Moving the plasma radially (DIII-D)
   - *Tight wall, only small movements allowed in ITER*

The third actuator is the only one possible in ITER:

3. Steering the launcher (ITER)



Steering of ECCD will feed back on measurements of ρ_{ECH} and ρ_{NTM}



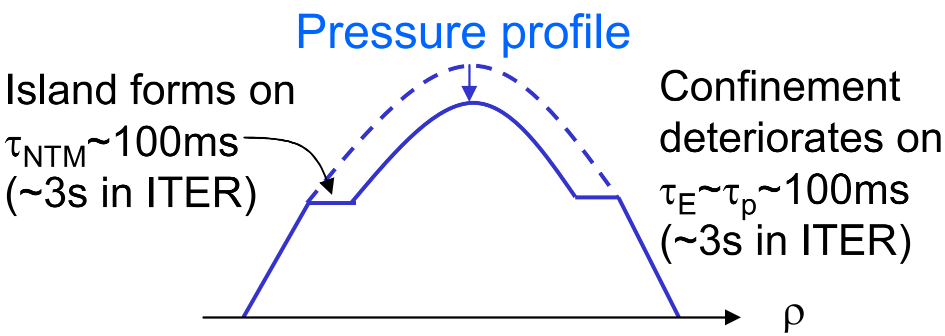
1: needs NTM

2: needs modulated ECH

3: doesn't need NTM → Use before NTM appears, for pre-emptive ECH/ECCD

4: doesn't need ECH → Decide if and when to fire ECH/ECCD for NTMs. Otherwise, ECH/ECCD can be used for: sawtooth control, ELM control, profile tailoring, ITBs, central heating, prevention of W accumulation…

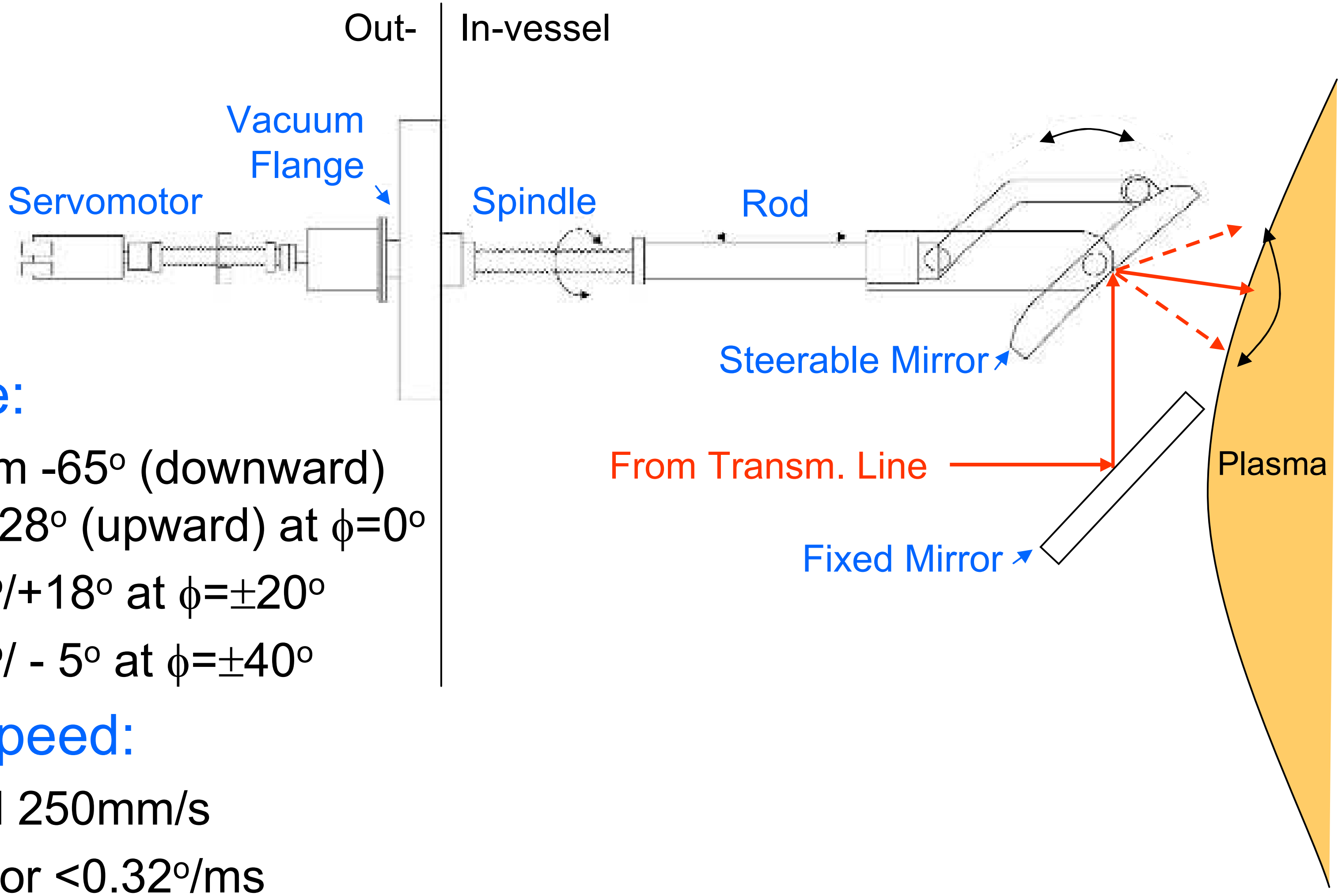
Real-time Steering: How fast do we *need* to respond? How fast *can* we respond?



- Physical Requirements:
  - NTM Physics
    - Growth ~100ms
    - Stabilization ~500ms
      (highly dependent on power and alignment)
  - Impact of NTM on Confinement ~100ms
  - Equilibrium Evolution & ECCD Aiming
    - Precision 2cm (Island width w=2-5cm, ECRH deposition $\delta_{ECCD} \approx 4$cm)
    - Rational surface can move by 2cm in (much) less than 100ms

- Engineering Constraints:
  - Electronic: servomotor control latency time=5-25ms (see slides 9, 10)
  - Mechanic: mirror inertia → max acceleration (see slides 8,10)

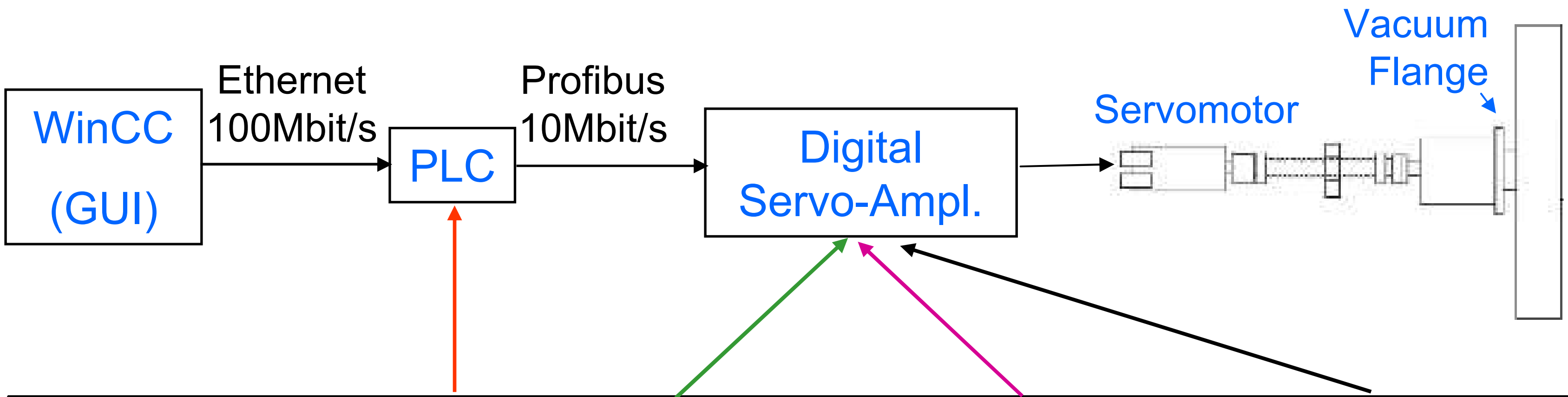
Push-Rod steers Mirror poloidally



- Range:
  - From -65° (downward) to +28° (upward) at ϕ=0°
  - -63°/+18° at ϕ=±20°
  - -60°/ - 5° at ϕ=±40°
- Max speed:
  - Rod 250mm/s
  - Mirror <0.32°/ms (up to 1cm/ms in plasma)

F.Monaco and H.Schütz

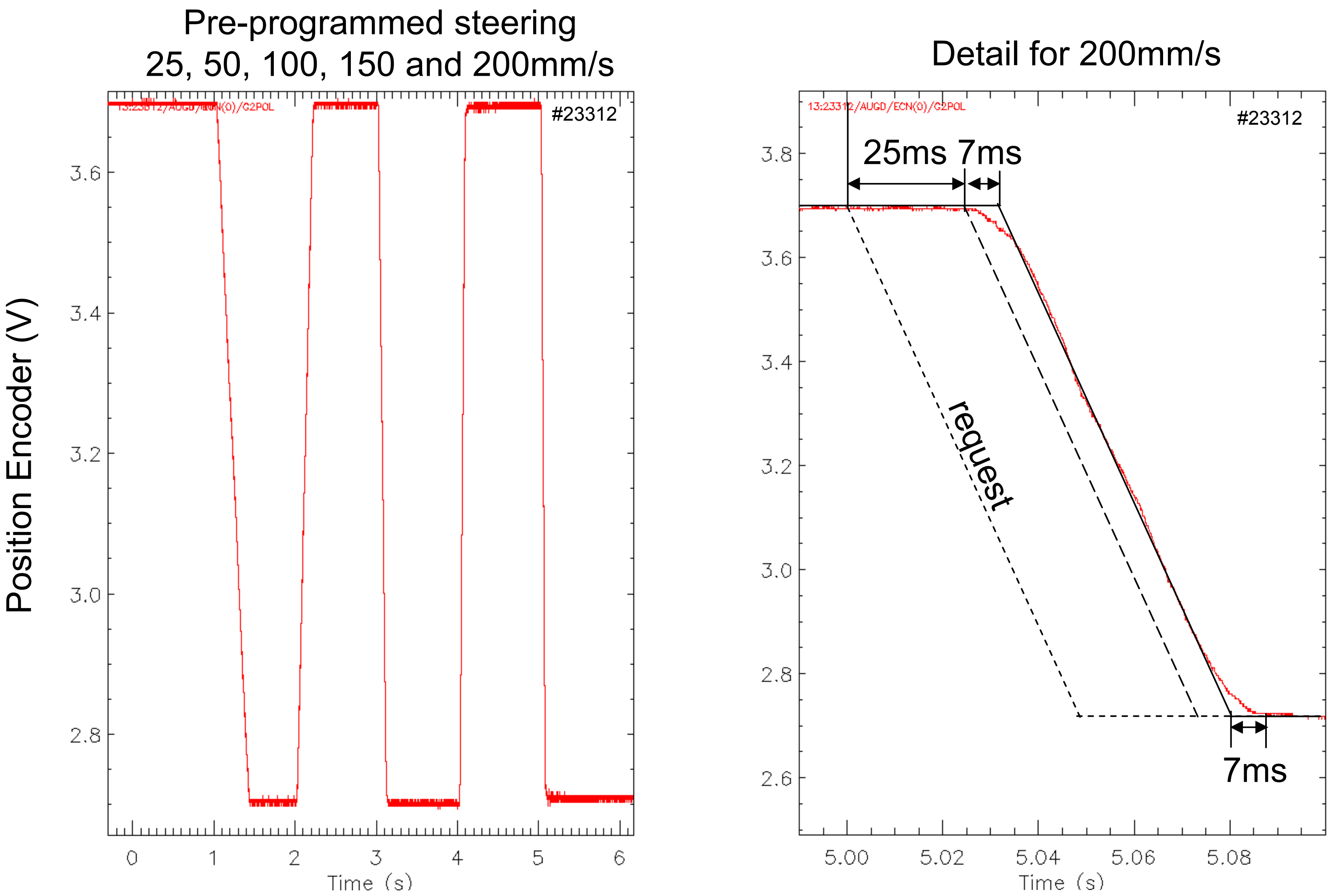
Servomotor can be controlled in real-time by 4 different methods. “Lower” level goes faster.



	1. Prescribe next position	2. Prescribe velocity for next step	3. Order positive, negative or no step (fixed length)	4. Order arbitrary nr.steps of fixed length, in arbitrary direction
Latency	21±3ms (3 cycles)	<5ms	12ms	7ms
Time resolution*			>33ms	>1ms
Comments	√	No reference. Sens. to Noise		Future work

* provided velocity<250mm/s i.e. 0.32°/ms

Latency, acceleration and deceleration were measured



Previous work: Look-Up Table (LUT)

- 3x triangularity δ=0, 0.2, 0.4
- 4x toroidal field B_T=-1.9, 2.1, 2.3, 2.5T
- 2x beta (→Shafranov shift) β_N=1, 3
- 3x vert.position $z_{mag.axis}$=0, 16, 32cm
- 1x elongation κ=1.7
- 4x central density n_{e0}=4, 6, 8, 10 $\cdot 10^{19} m^{-3}$
- 2x density peaking H1/H5=1.4, 1.8
- 5x launch angle, pol θ=-15, -7, 10, 20°
- 2x launch angle, tor ϕ=-10, 0°

=5760 ray tracing results in data base

Adriano Manini

Comprehensive Look-Up Table would be huge and hard to interpolate

- More input values needed, and non-uniformly distributed, as ray response is non-linear :
  - 3x triangularity δ=0, 0.2, 0.4
  - 6x toroidal field B_T=-1.9, 2.1, 2.2, 2.3, 2.4, 2.5T
  - 3x beta (→Shafranov shift) β_N=1, 2.2, 3
  - 5x vert.position $z_{mag.axis}$=0, 8, 16, 24, 32cm
  - 2x elongation κ=1.5, 1.7
  - 8x central density n_{e0}=4, 6, 8, 10, 11, 11.5, 12, 12.5 $\cdot 10^{19}$m^{-3}
  - 2x density peaking H1/H5=1.4, 1.8
  - 6x launch angle, pol. θ= -22, -15, -7, 0, 10, 20^o
  - 2x launch angle, tor. ϕ=-10, 0^o

  Overdense core, but mid-radius accessible

  =103680 ray tracing results in data base

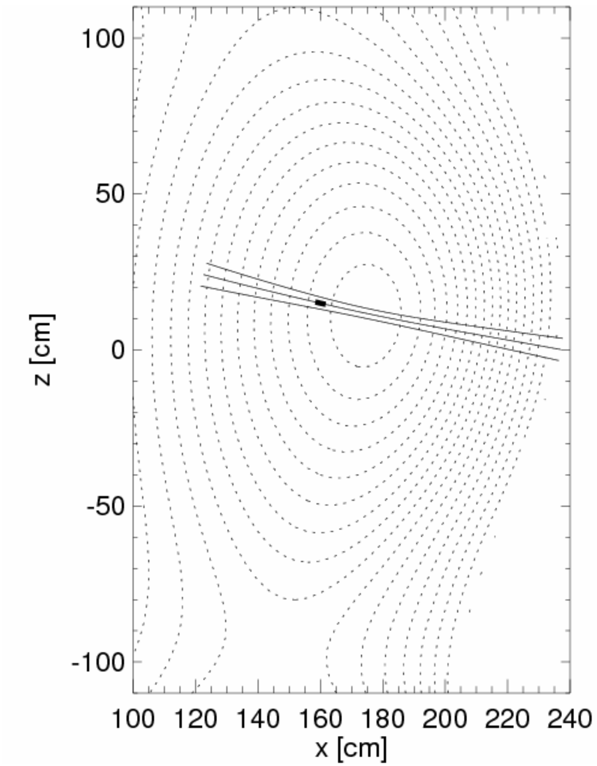
- Loss of information in numerical (arbitrary shape) → analytical (δ, κ) equilibrium

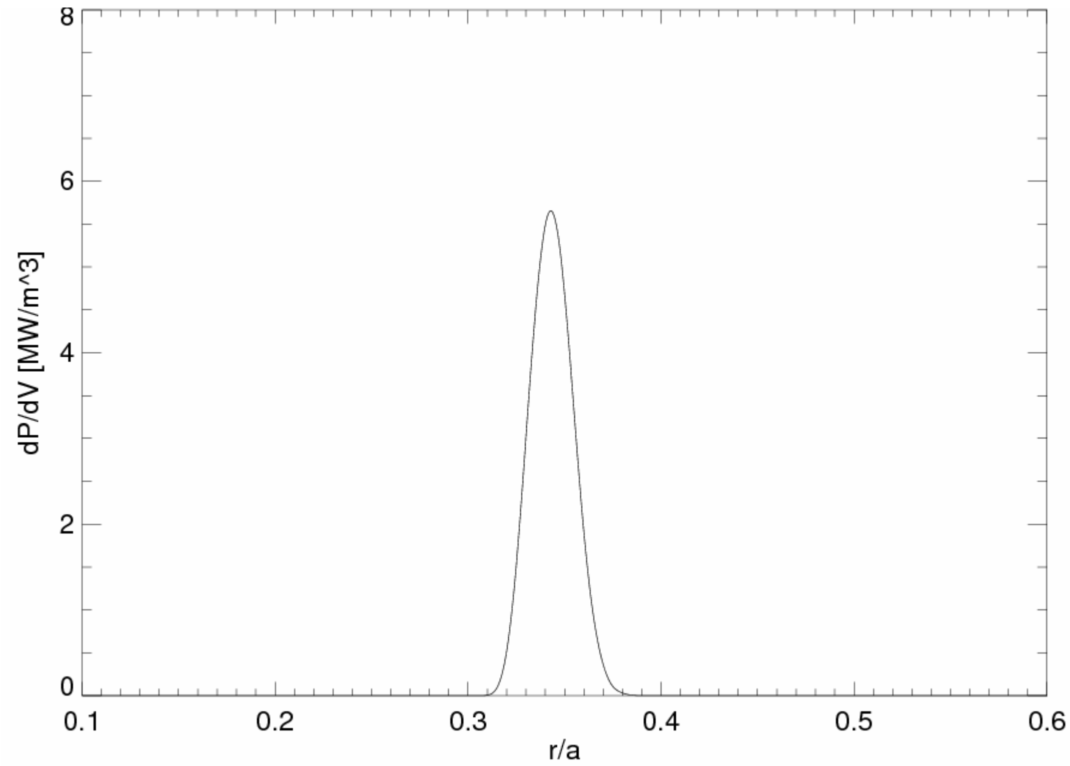
Combination of Physics and Programming made TORBEAM 20 times faster

1400ms	Original TORBEAM, on p01.bc (Linux pc, Intel Xeon 3GHz)
1400ms	Translated in f90 and made modular (was a single 14000 lines file)
1090ms	Removed repetitive if-then-else from interpolators of equilibria and profiles, unnecessarily checking input (minor radii) to be monotonic.

Combination of Physics and Programming made TORBEAM 20 times faster (cont'd)

540ms Omitted calculations of volumes between adjacent flux surfaces. Affects normalization, but not shape, of dP/dρ and $dj_{ECCD}/d\rho$ profiles. ρ of maximum j_{ECCD} still reliable.





Combination of Physics and Programming made TORBEAM 20 times faster (cont'd)

385ms	Reduced set of equations from beam tracing (18+1) to ray tracing (6+1). Beam width and wavefront curvature affect CD width but not CD peak
217ms	Equilibria and profiles are piecewise polynomially interpolated. New 5th order polynomials are 0.1% less precise but 49/25 times faster than old, 7th order polynomials
128ms	Omitted print-out on screen and files
	Ray tracing stopped when power $=10^{-9}$ of initial value
78ms	changed compiler from f95f to f95i, with optimizations
71ms	Relaxed tolerances in Ordinary Differential Eq. Solver (ray tracing only valid on scales $>2\text{-}3\lambda$)
66ms	architecture-specific (Pentium 4) optimizations
20ms?	migration to 4 core 64 bit processor, in progress

Additional Physics and Programming might make TORBEAM even faster

- In vacuo just use Geometrical Optics
- Calculate absorption only when likely ≠0 (i.e. when $\omega - n\omega_{ce} < 3k_{||}v_{th,e}$)
- Initial Equilibrium Interpolations can be
  - either speeded up
  - or separated and run in parallel
  - or avoided altogether: direct read from L.Giannone's Labview

Look- up Table will benefit from fast TORBEAM, and vice versa

- Less time to build the Look-up Table (LUT)
- Systematic study of sensitivity of ray tracing results to input (n_e, T_e, launch angles, equilibrium etc.) will help constructing smart LUT and using it in combination with TORBEAM:
  - If input varied negligibly →do nothing
  - If input varied modestly → consult LUT locally and make small corrections to latest TORBEAM results
  - If input varied significantly, re-run TORBEAM

- TORBEAM results can be saved in LUT "on the go" and be interpolated later in the same shot or in future shots.
  - continuously growing or "learning" data-base
  - data-base denser where more useful, i.e. where more experiments
  - similar to disruption data base for Neural Network "training"

ASDEX Upgrade

Summary and Conclusions

- Real-time mirror steering is the only ITER-relevant way of radially aligning ECCD to NTMs
- NTM growth rate requires real-time control on <100ms time-scale, at AUG
- Mirror inertia and electronic latency set technical limit to ~35ms
- A fast version of TORBEAM was realized, which calculates ρ of peak ECCD every 66ms


Torbeam
Torbeam
IPP
Shell
AMD
Vodafone
Marlboro
BRIDGESTONE
66ms

Iterative parabolic interpolation of 9D subarray