

# Learning Feedforward for repetitive disturbances

Christian Schmidt

Deutsches Elektronen Synchrotron

Technische Universität Hamburg Harburg

FLASH Seminar

May 8, 2009

# Outline

- System description and modeling
  - Disturbance sources
  - Excitation procedure
  - Model verification
- Iterative learning control
  - Implementation
  - Measurement results
  - Future application
- Conclusion and Outlook

# Schematic view of the Control System

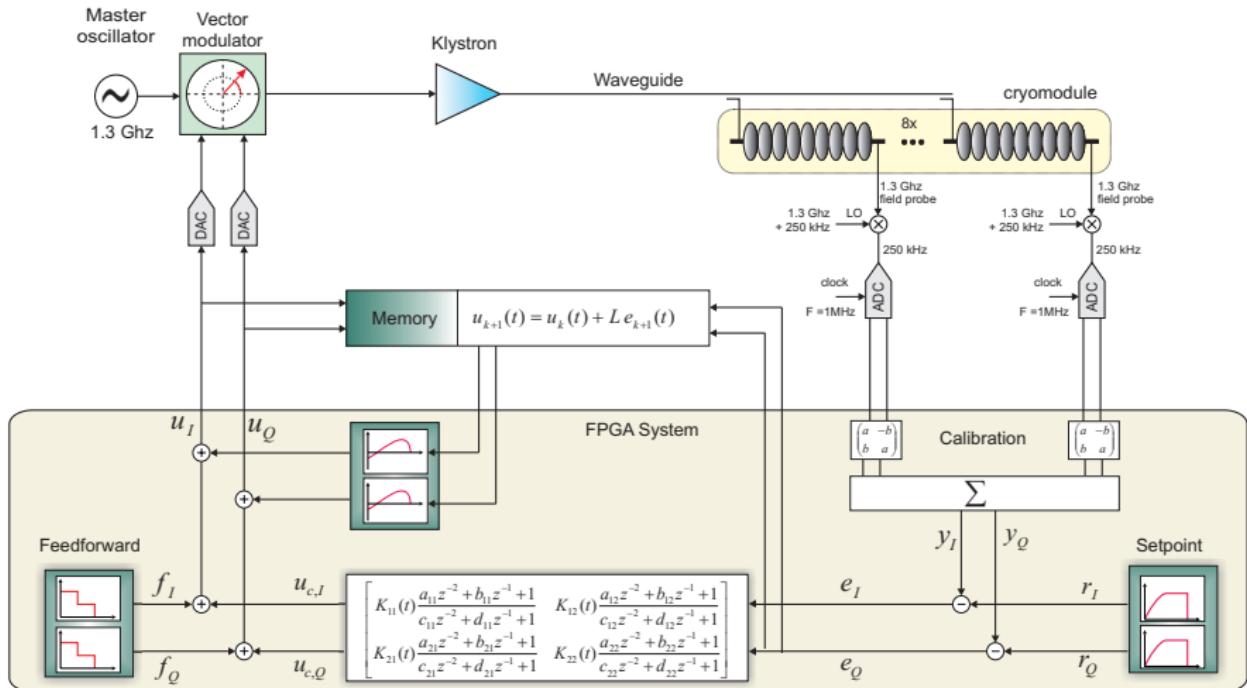
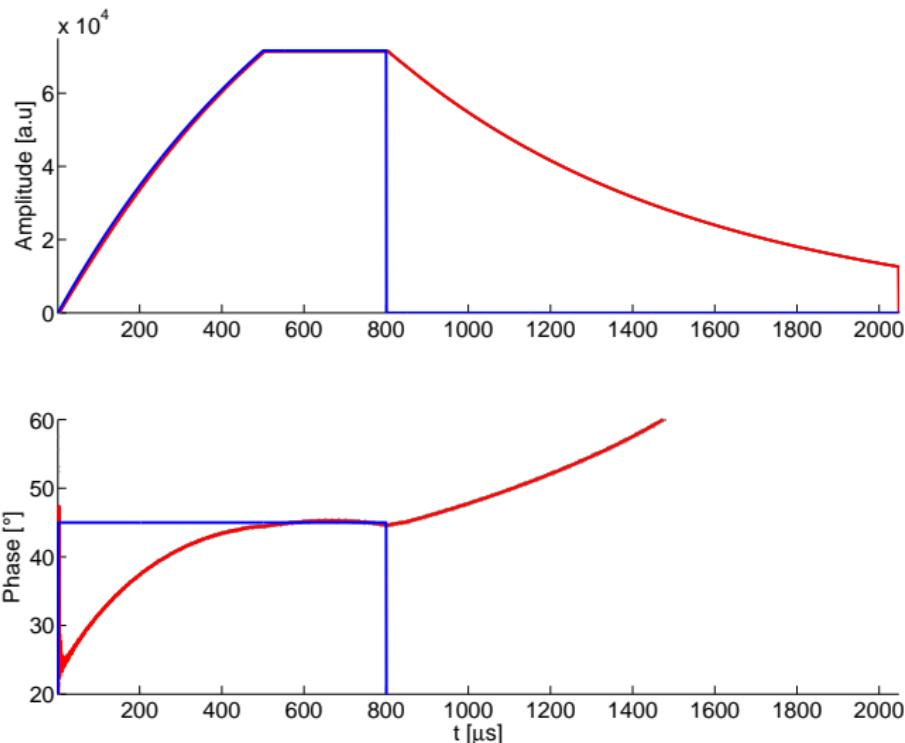


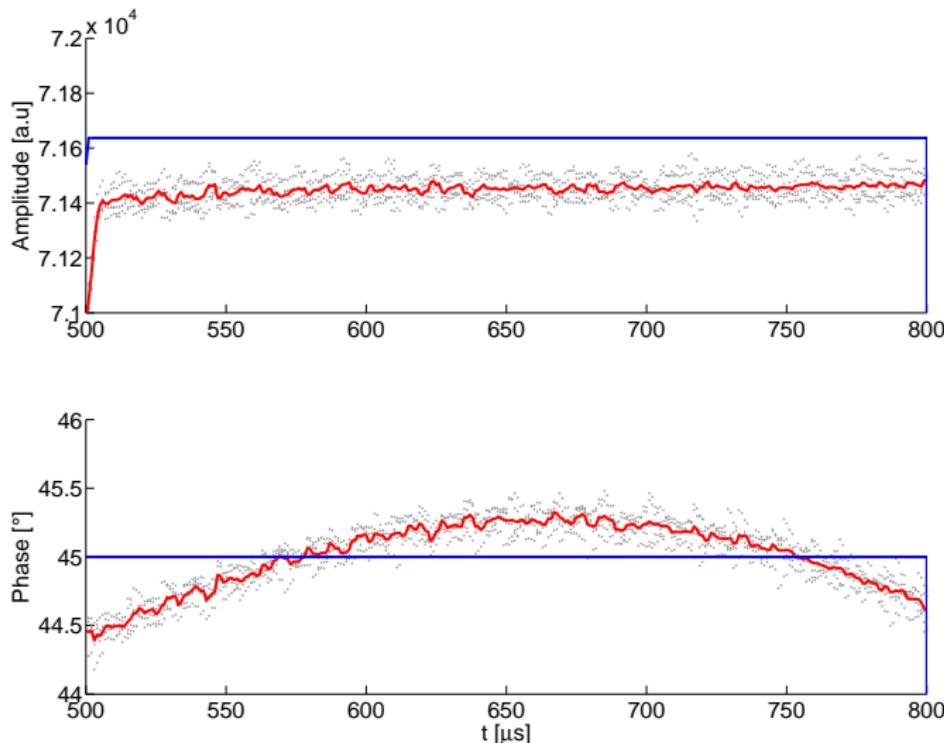
Figure: Controller structure and signals

# Open Loop measurements with feedforward



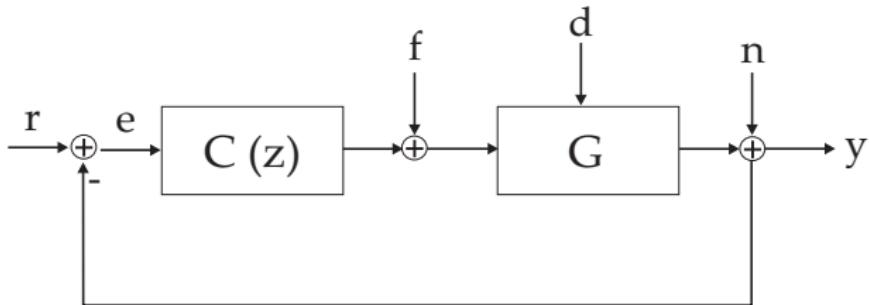
- Detuning effects, pulse to pulse fluctuations

# Open Loop measurements with feedforward



- Detuning effects, pulse to pulse fluctuations

# Classification of disturbance sources

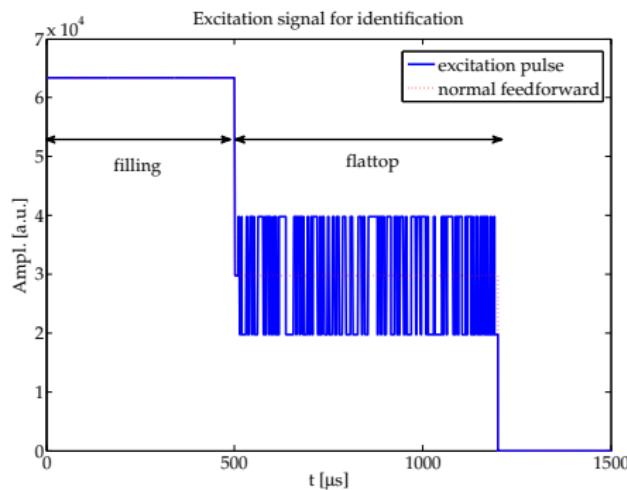


$f$	$d$	$n$
klystron waveguide noise	microphonics LF detuning phase drifts	field detection downconversion discretization

- FB controller for field stabilization and noise suppression
- Repetitive effects  $\Rightarrow$  adaptation of feedforward signal

# System identification and modeling

- Appropriate model is needed for controller design
- Excitation of the system to model dynamic behavior
- Estimating black box model for current operation point
- Limitations due to flattop length

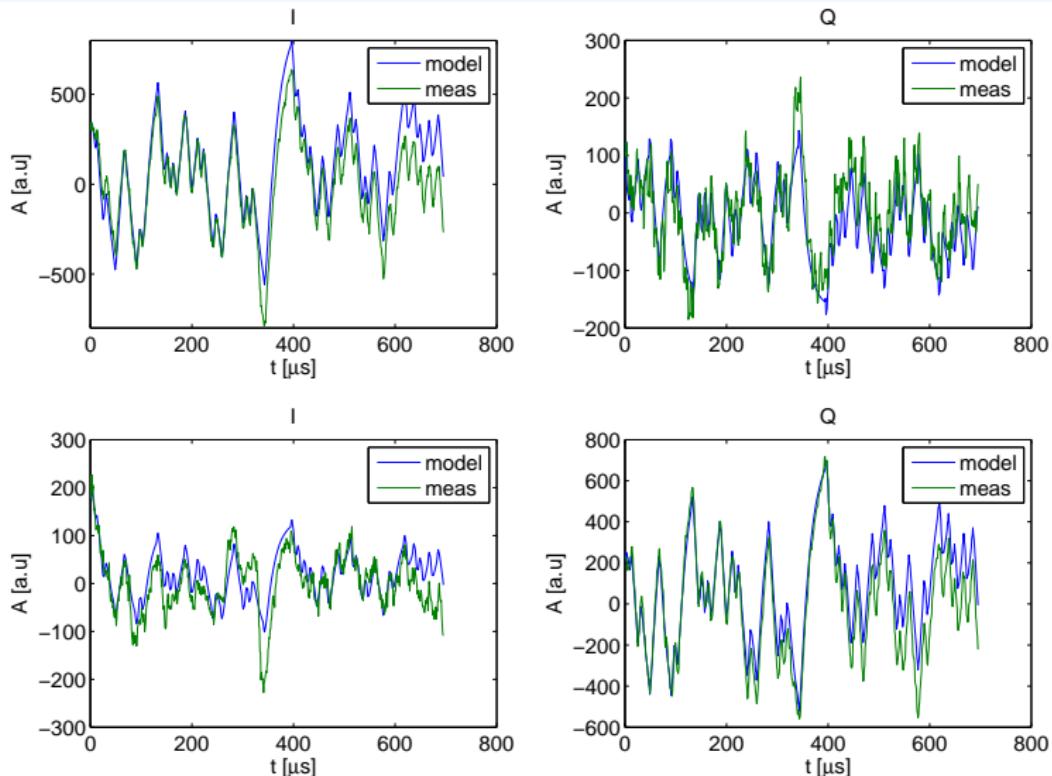


- State Space system (LTI)

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}$$

- Estimating system parameters  $A, B, C, D$  with subspace algorithm n4sid

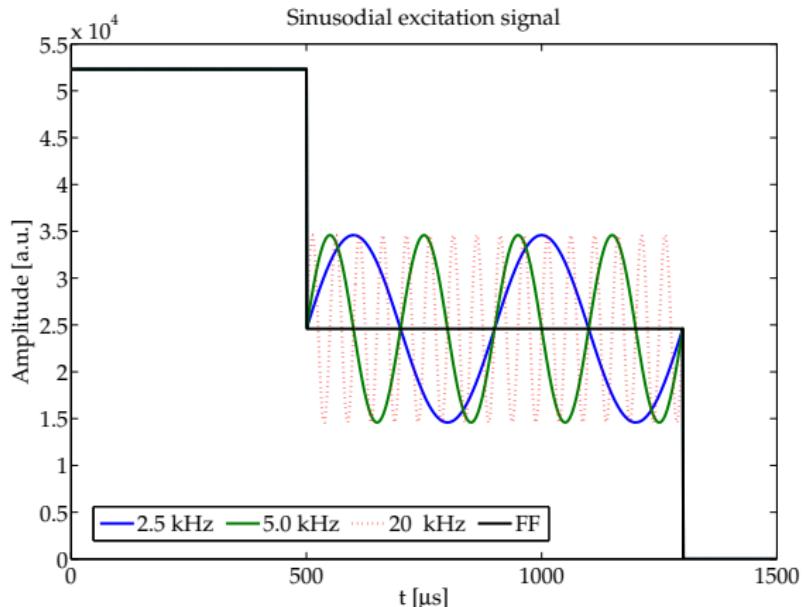
# Model validation with measurement data



- Appropriate model of low frequency system dynamics

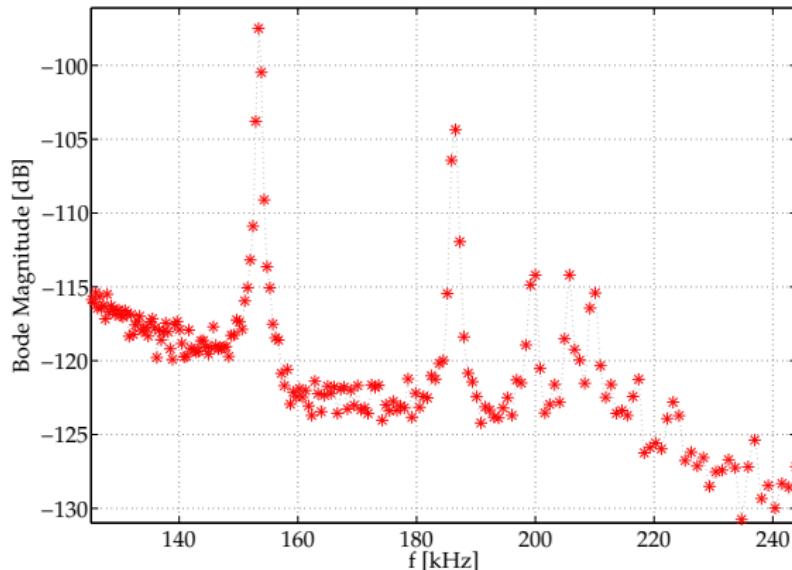
# Excitation with stable oscillations

- Sweeping frequency sin-waves
- Stable uniform excitation frequency signal
- Searching for resonance peaks



# Zoomed region of response

Sinusodial Excitation during Flattop



- ↗ Combination of different excitation signals
- More precise model for higher frequency range

# Iterative learning control

**Idea:** Use information from previous trials to improve the FF signal for upcoming pulses.

**Issue:** Only repetitive signals can be covered!

General form of input update equation:

$$u_{k+1} = Q(u_k + L e_k) \quad k \Rightarrow \text{number of trial}$$

Norm-Optimal Iterative learning Control<sup>3</sup>

$Q \rightarrow$  Identity matrix

$L \rightarrow$  Non causal, time varying, model based filter

- Computation between pulses

---

<sup>3</sup> N. Amann, D.H. Owens and E. Rogers, *Iterative learning control for discrete-time systems with exponential rate of convergence*, IEEE Proceedings - Control Theory and Applications, 217-223, 143, 1996

# Current implementation

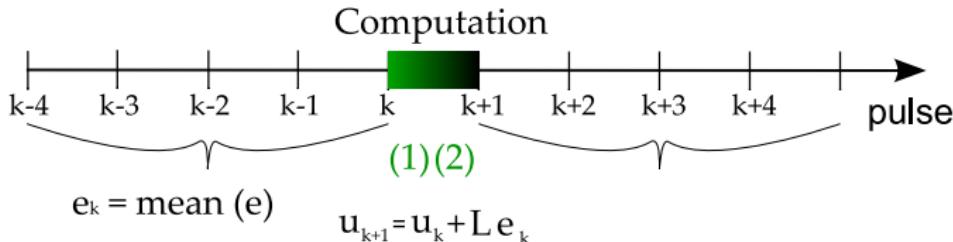
- Currently using Matlab for computation
- Only RF information is taken for adaptation

Between trials this update equations must be calculated:

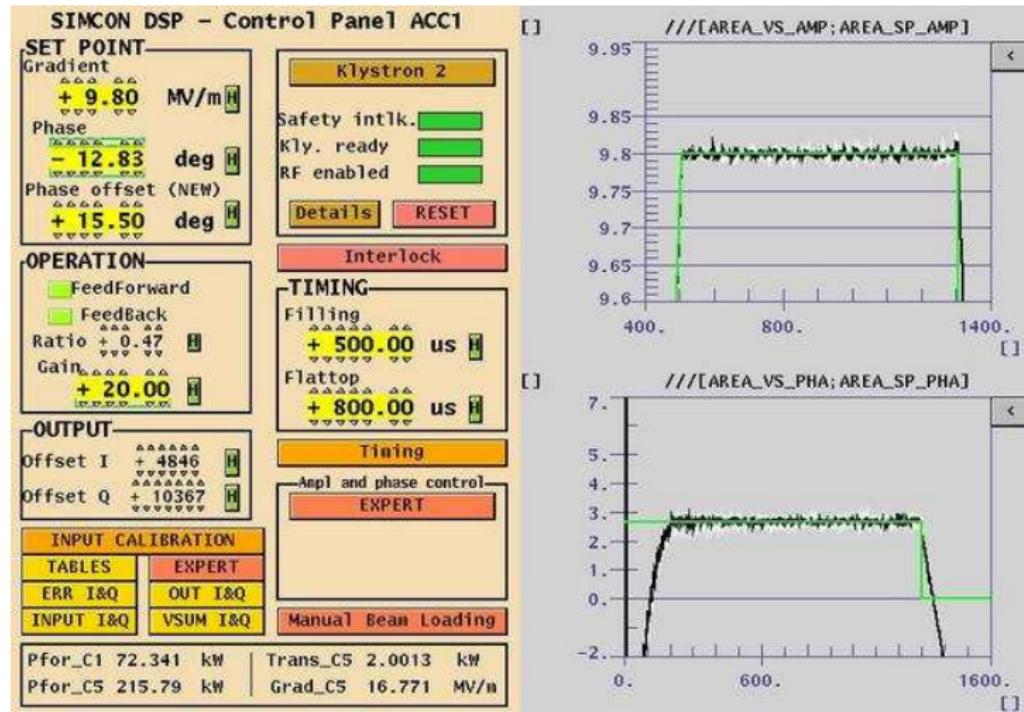
$$\xi_{k+1}(t-1) = \beta \xi_{k+1}(t) + \gamma e_k(t) ; \quad \xi(T) = 0 \quad (1)$$

$$u_{k+1}(t) = u_k(t) + R^{-1} B^T \xi_{k+1}(t+1) \quad (2)$$

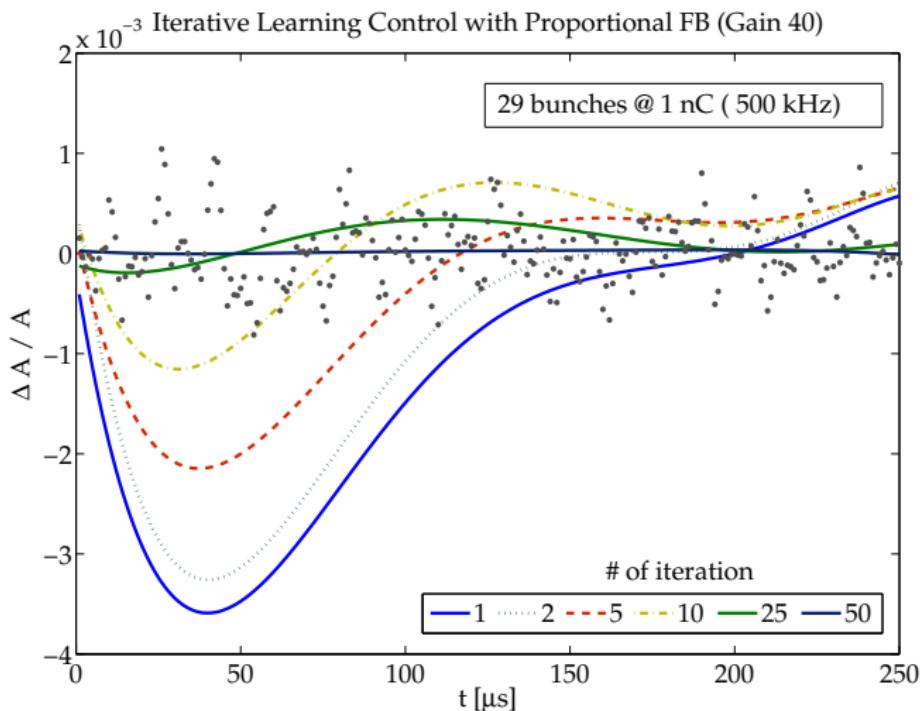
$\beta$  ;  $\gamma$   $\Rightarrow$  update matrices are computed before trials



# RF field after adaptation

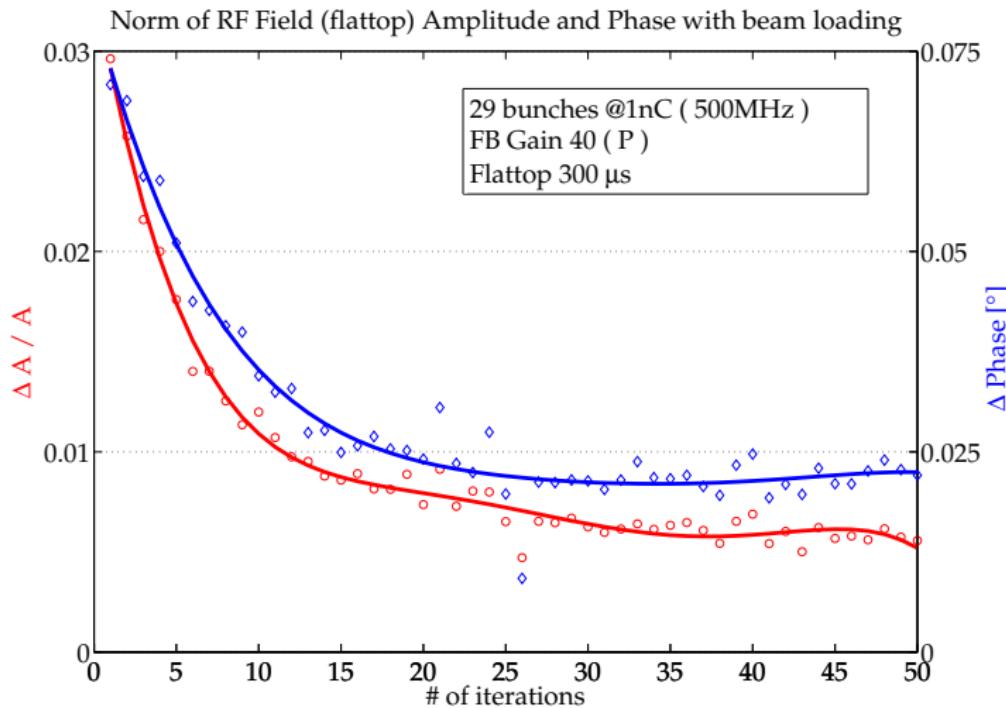


# Amplitude correction with learning FF



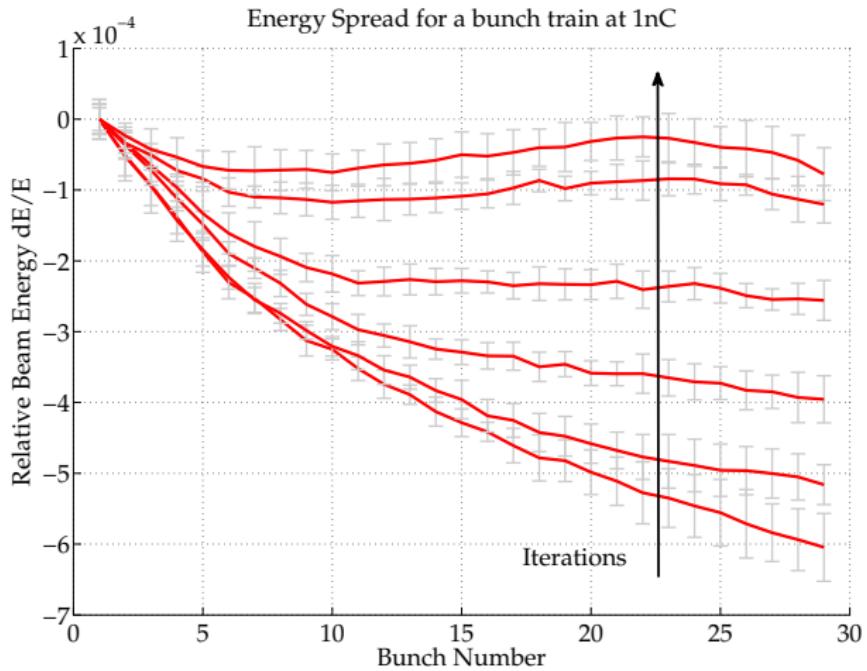
- Beam induced transient is compensated by FF adaptation
- Feedback controller was turned on (GAIN 40)

# Removing deviation errors



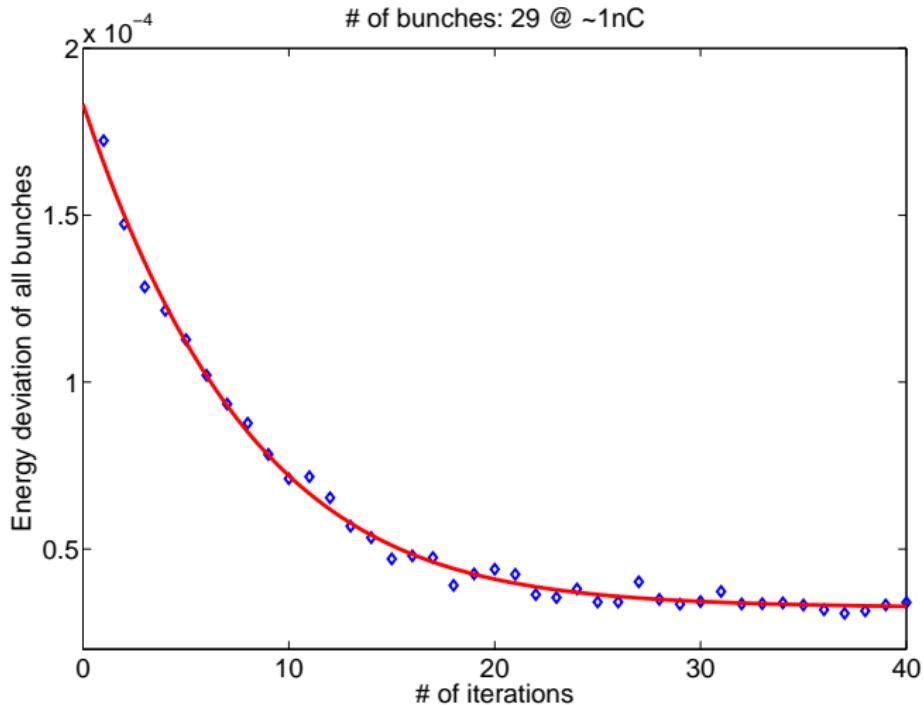
- Norm of RF amplitude and phase errors during flattop
- Fast convergence and stable

# Measurements with beam



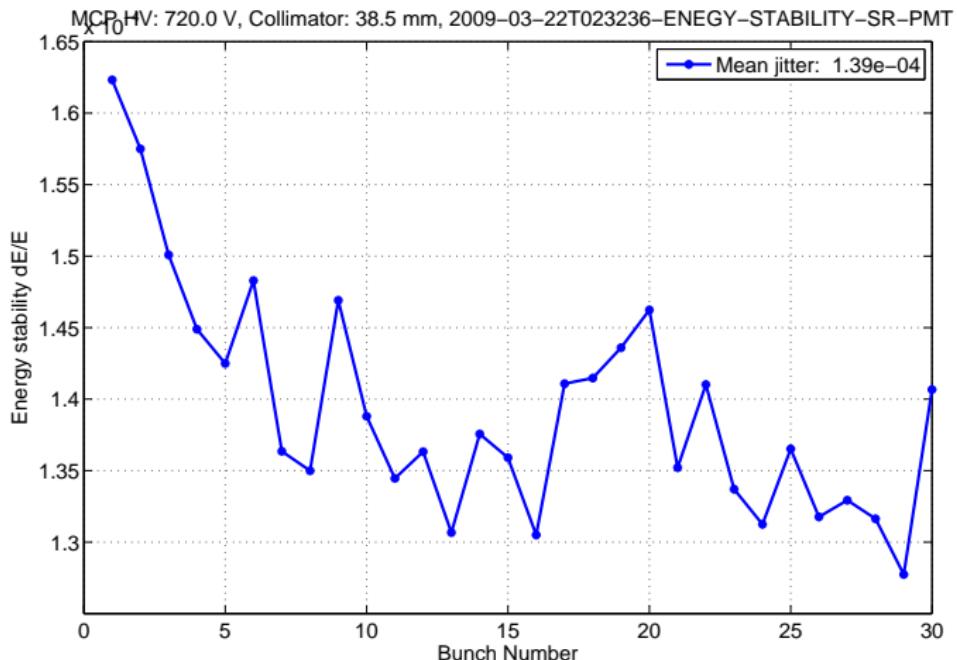
- Energy spread reduced within 50 iterations
- Pulse to pulse fluctuations are mostly stochastic

# Convergence of beam energy



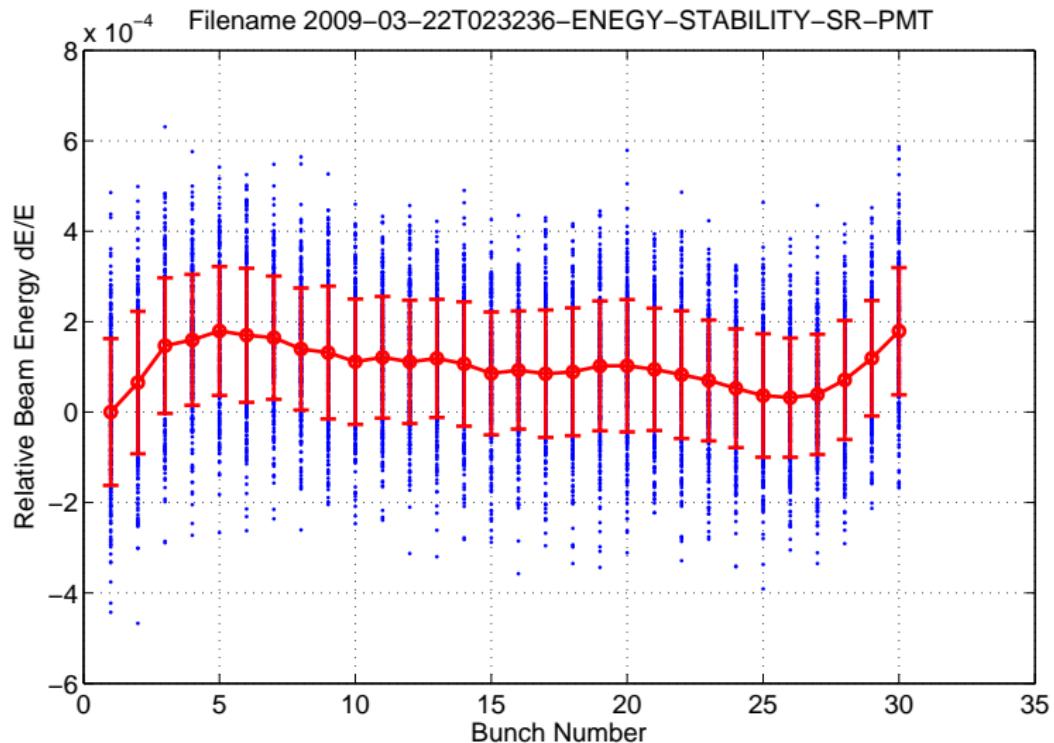
- Norm of beam energy spread over bunch train

# Energy stability spread over 30 bunches



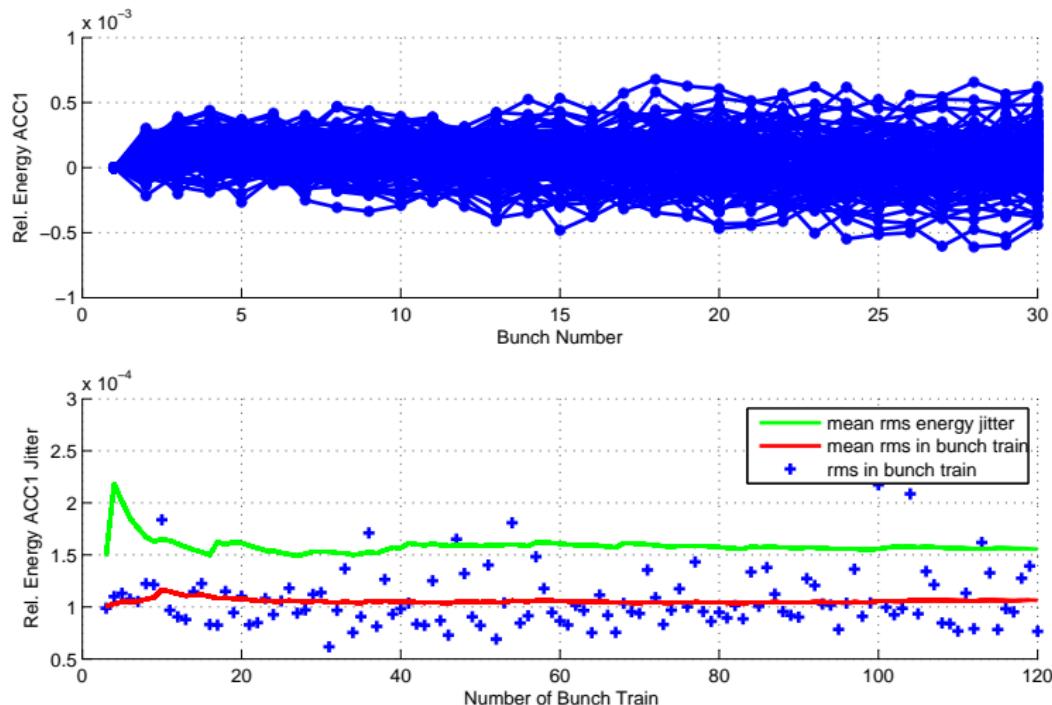
Thanks to C. Gerth

# Relative beam energy deviation



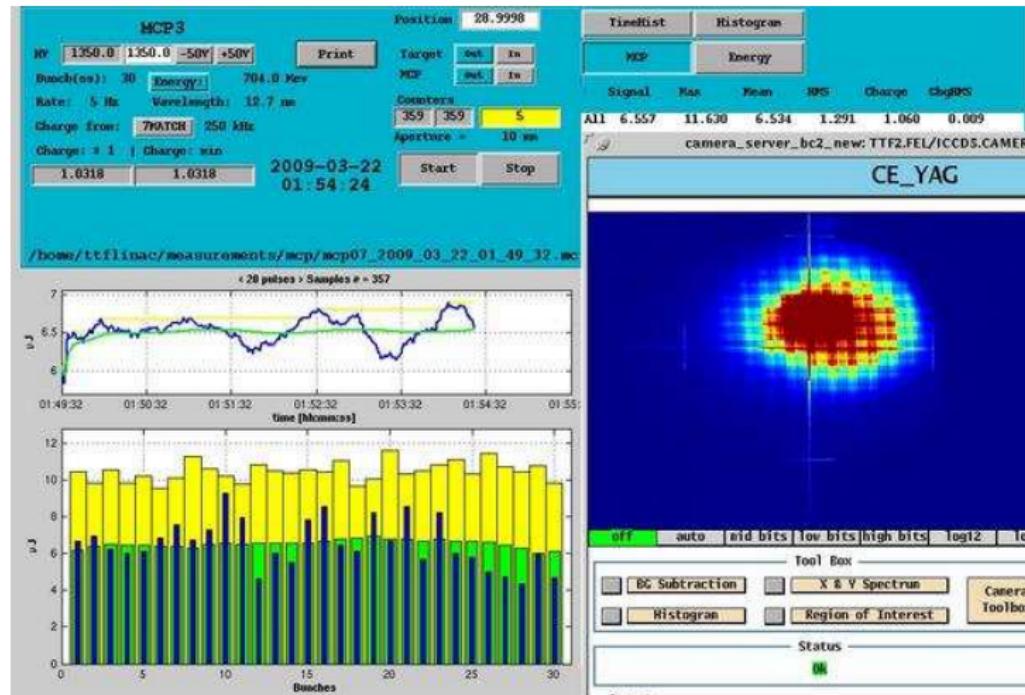
Thanks to C. Gerth

# Pulse to pulse energy jitter



Thanks to C. Gerth

# SASE after adaptation



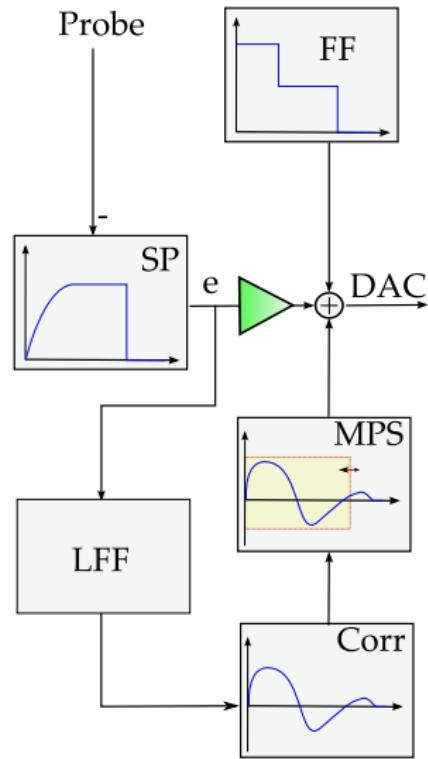
# Implementation of Learning FF

Deterministic signals to be covered

- Beam loading !
- Lorentz force detuning
- Filling to flattop transition

Exception handling for permanent implementation

- Protection against quenches in absence of beam
  - Software issues
  - Fast recoverable / convergence
- ⇒ Fast windowing correction tables



# Summary

- Identification procedure works sufficient
- Iterative Learning Control algorithm implemented with Matlab
- Compensation of beam loading using rf field adaptation
- Measurements of energy spread over bunch train
- Fast convergence and stable to changes in setpoint / disturbances
- Fast implementation using C++ ( pulse to pulse )
- Combination with feedback controller + Beam information
- Automation !

Special thanks to:

G.Lichtenberg, C. Gerth, F.Ludwig