11th International Computational Accelerator Physics Conference – ICAP 2012

Simulations of Multibunch Instabilities with the HEADTAIL Code

Nicolas Mounet, Elias Métral and Giovanni Rumolo

Acknowledgments: G. Arduini, R. Assmann, R. Bruce, W. Höfle, V. Kain, L. Ponce, L. Rivkin, A. Rossi, B. Salvant, D. Valuch, D. Wollmann, B. Zotter.

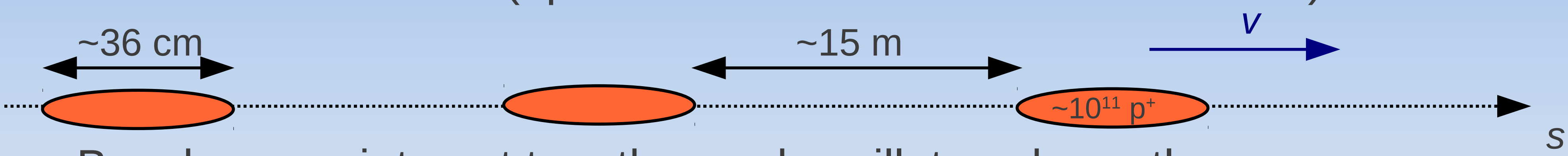
Simulation of Multibunch Instabilities with the HEADTAIL Code

- Introduction: coupled-bunch (or multibunch) instabilities
- Preliminary: on the calculation of wake functions
- Presentation of HEADTAIL multibunch
- Benchmarks with theory
- Simulation of coupled-bunch instabilities for the CERN LHC

Coupled-bunch instabilities

Coupled-bunch instabilities

- For bunched beams (up to 1380 bunches in the CERN LHC)

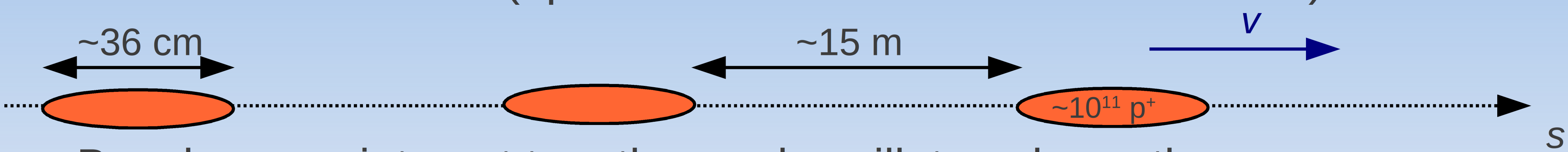


Bunches can interact together and oscillate coherently.

Example with 36 bunches in the LHC: oscillation pattern along the bunch train (simulation result):

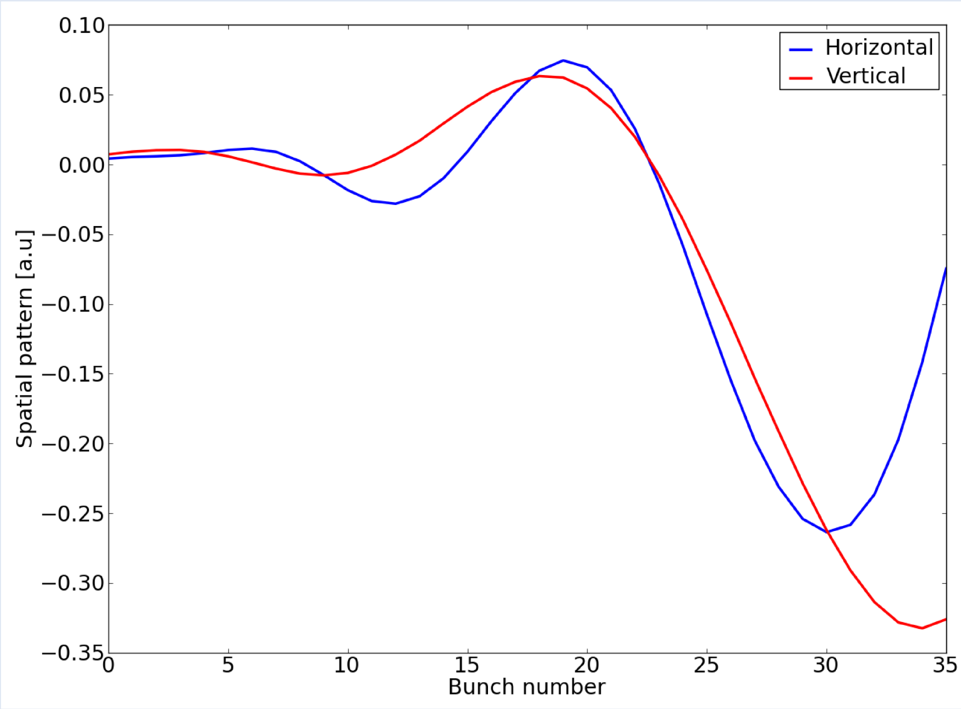
Coupled-bunch instabilities

- For bunched beams (up to 1380 bunches in the CERN LHC)



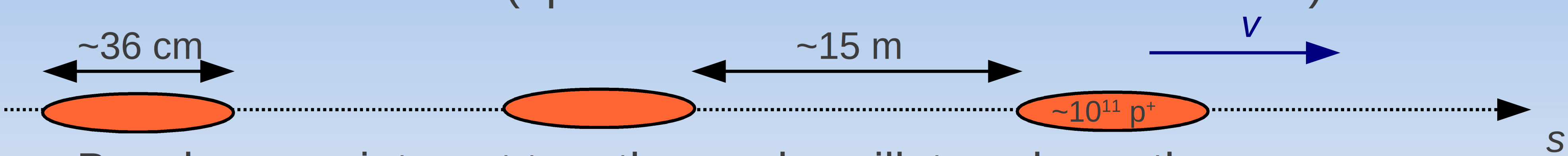
Bunches can interact together and oscillate coherently.

Example with 36 bunches in the LHC: oscillation pattern along the bunch train (simulation result):



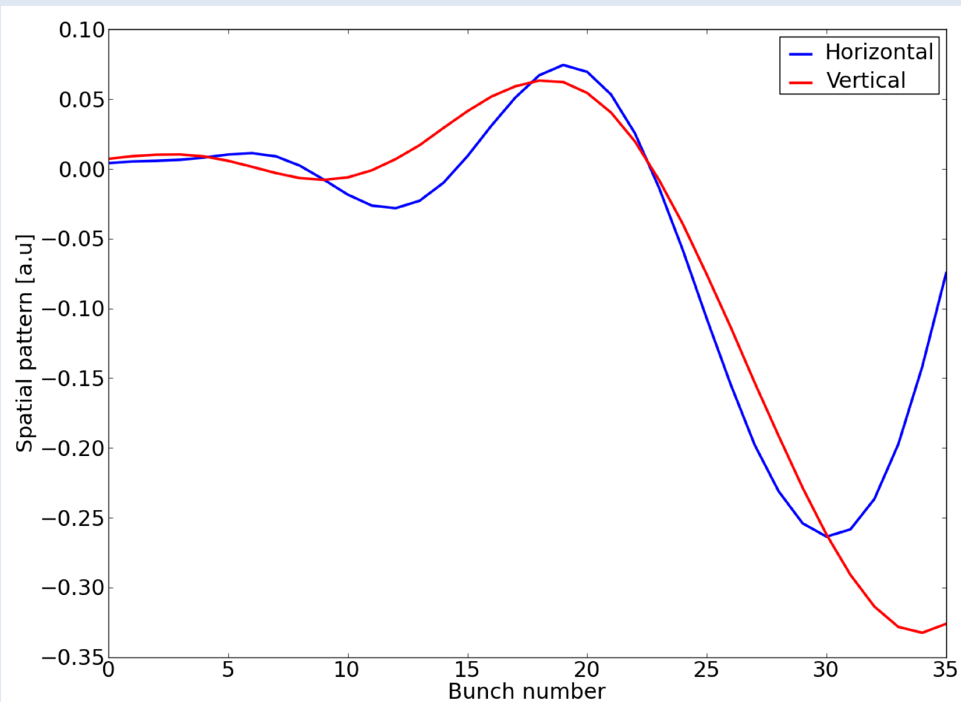
Coupled-bunch instabilities

- For bunched beams (up to 1380 bunches in the CERN LHC)



Bunches can interact together and oscillate coherently.

Example with 36 bunches in the LHC: oscillation pattern along the bunch train (simulation result):



→ **Coupled-bunch instabilities**

- Must be damped by feedback system and/or Landau damping.
- Need to study them to know if damping mechanisms are sufficient.

Before launching HEADTAIL: how to obtain wake functions ?

Before launching HEADTAIL: how to obtain wake functions ?

- HEADTAIL is a time domain code → requires wake functions.
- These can be obtained from 3D electromagnetic fields simulations, but

→ can be long if we have to do this for all elements of a machine (collimators, beam pipe with different cross-sections, etc.),

→ long range behaviour (longer than bunch spacing or circumference) would require huge number of mesh cells.

Before launching HEADTAIL: how to obtain wake functions ?

- HEADTAIL is a time domain code → requires wake functions.
- These can be obtained from 3D electromagnetic fields simulations, but

  → can be long if we have to do this for all elements of a machine (collimators, beam pipe with different cross-sections, etc.),

  → long range behaviour (longer than bunch spacing or circumference) would require huge number of mesh cells.
- Simple analytic formulas exist but very limited range of application.

Before launching HEADTAIL: how to obtain wake functions ?

- HEADTAIL is a time domain code → requires wake functions.
- These can be obtained from 3D electromagnetic fields simulations, but

  → can be long if we have to do this for all elements of a machine (collimators, beam pipe with different cross-sections, etc.),

  → long range behaviour (longer than bunch spacing or circumference) would require huge number of mesh cells.
- Simple analytic formulas exist but very limited range of application.
- On the other hand, wake functions are the Fourier transforms of the impedances:

  $$W_x(\tau) = -\frac{j}{2\pi}\int_{-\infty}^{\infty} \mathrm{d}\omega e^{j\omega\tau} Z_x(\omega)$$ for a test particle τ seconds behind the source

  ⇒ impedances can be obtained analytically for simple (multilayer) 2D structures: **cylindrical** (Zotter 1969, Burov-Lebedev 2002, Zotter 2005, Wang-Qin 2007, Ivanyan et al 2008, Hahn 2010, Mounet-Métral 2010), **flat** (Piwinski 1984, Yokoya 1993, Burov-Lebedev 2002, Mounet-Métral 2010) or **elliptical** (Palumbo-Vaccaro 1985, Yokoya 1993, Piwinski 1994).

Short and long range wake functions

Short and long range wake functions

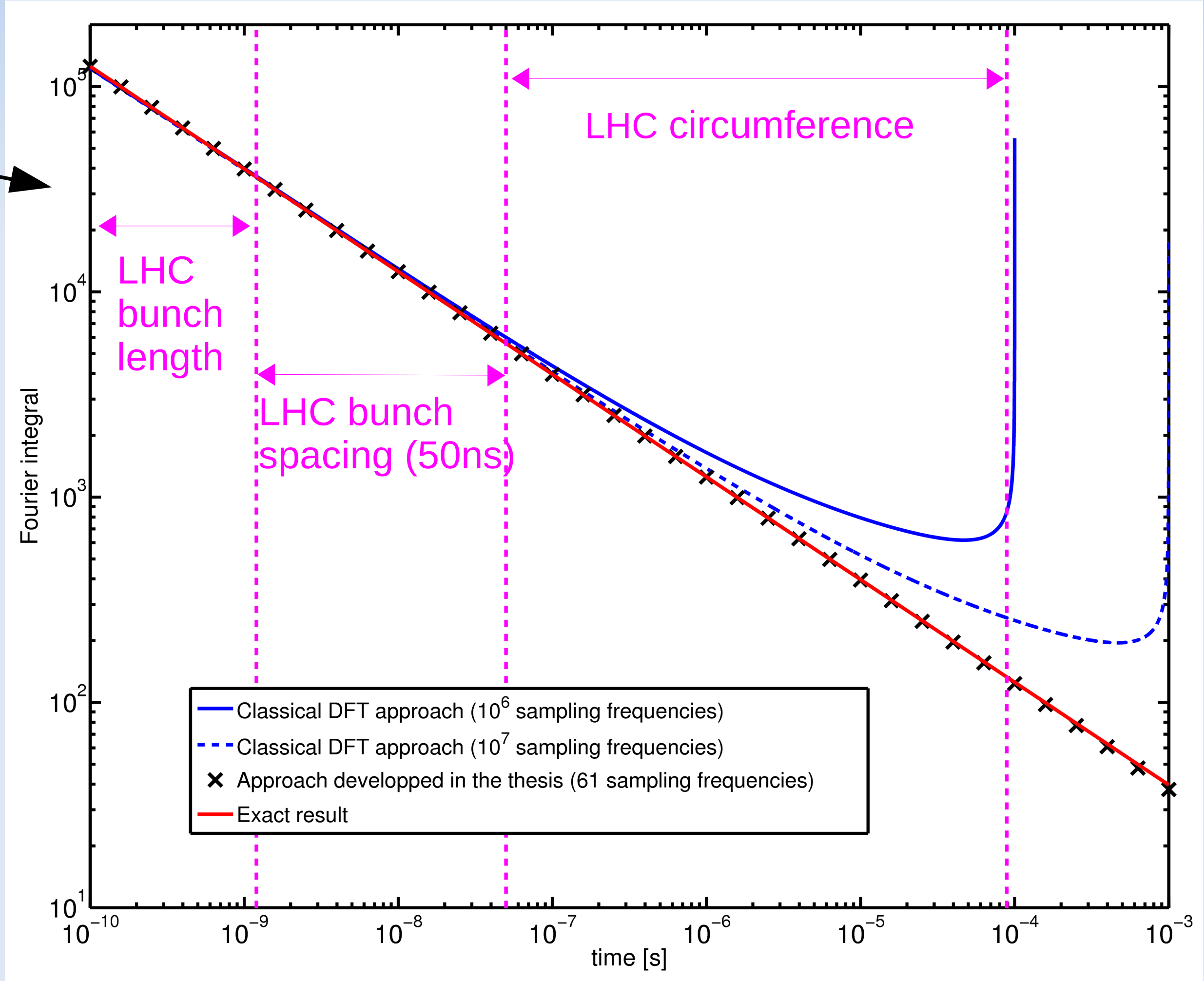
- Need wake functions both on short scale (intrabunch) and long scale (bunch spacing, even circumference) → FFT of impedances with evenly spaced frequency mesh not accurate enough when dealing with large frequency range.

Short and long range wake functions

- Need wake functions both on short scale (intrabunch) and long scale (bunch spacing, even circumference) → FFT of impedances with evenly spaced frequency mesh not accurate enough when dealing with large frequency range.

Example with $Z_x = 1/\sqrt{|\omega|}$

→ clearly DFT fails on part of the range of interest

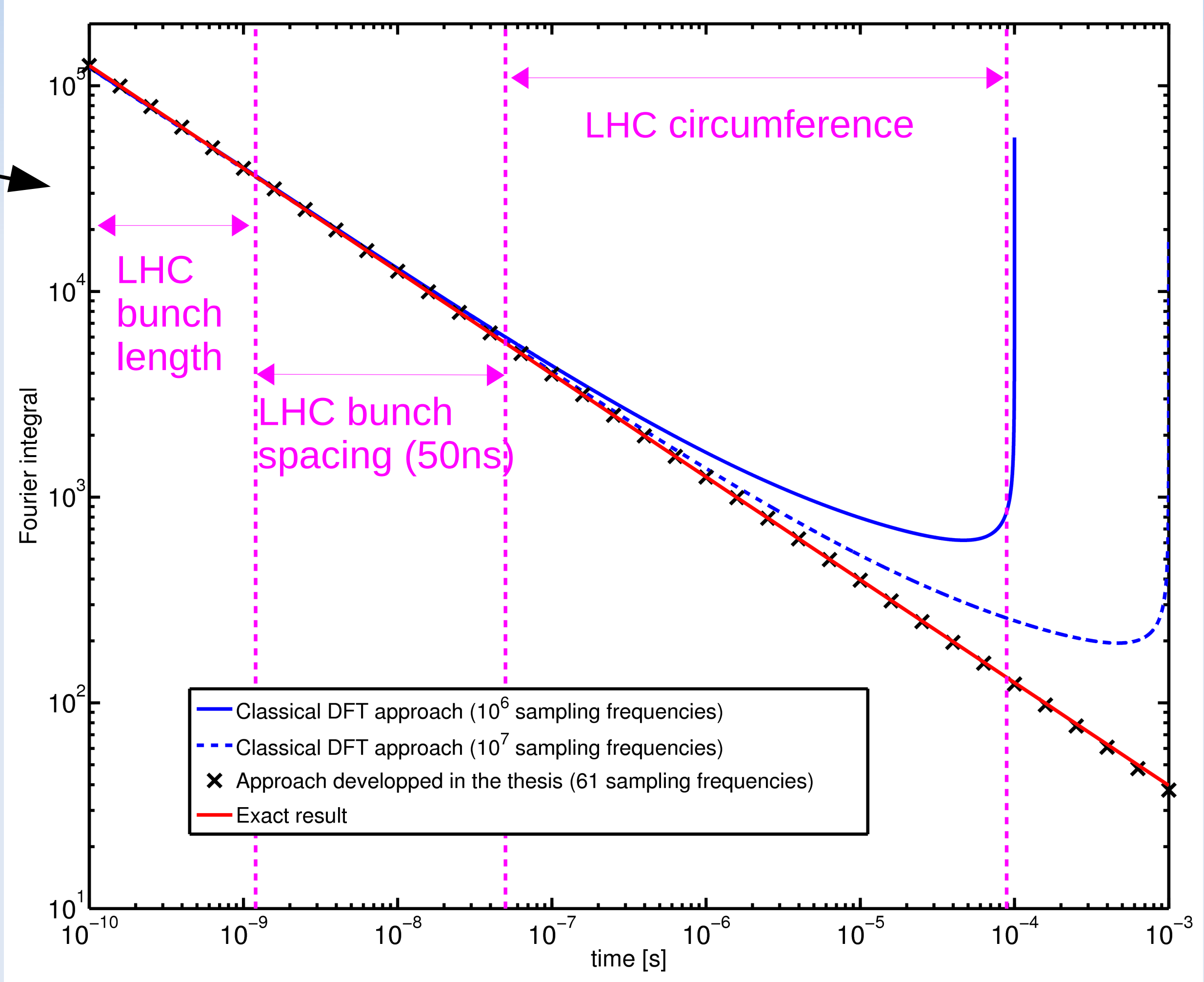


Short and long range wake functions

- Need wake functions both on short scale (intrabunch) and long scale (bunch spacing, even circumference) → FFT of impedances with evenly spaced frequency mesh not accurate enough when dealing with large frequency range.

Example with $Z_x = 1/\sqrt{|\omega|}$

→ clearly DFT fails on part of the range of interest

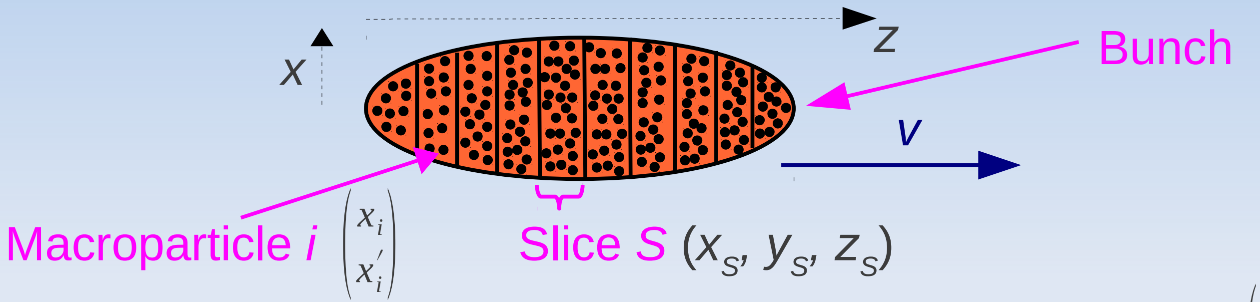


⇒ Use a "new" method (based on Filon's idea – 1928): given any frequency sampling, on each subinterval replace the impedance by its interpolation, and integrate it analytically.

→ more accurate and faster.

HEADTAIL Multibunch

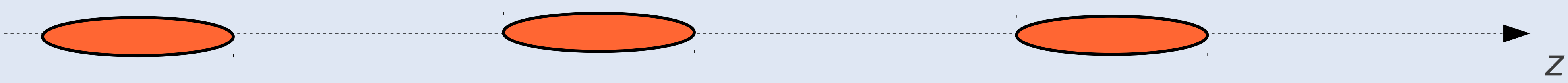
- HEADTAIL: beam dynamics simulation code, using macroparticles
  - Pre-existing single-bunch version (*G. Rumolo et al*, PRST-AB, 2002):



Each turn:
- macropart. *i* receives kick from the wake of all preceding slices: $\begin{pmatrix} x_i \\ x_i' \end{pmatrix} \rightarrow \begin{pmatrix} x_i \\ x_i' + \Delta x_i'(x_S, x_{S_i}, z_S - z_{S_i}) \end{pmatrix}$
- then it is transported through the machine lattice: $\begin{pmatrix} x_i \\ x_i' \end{pmatrix} \rightarrow M \cdot \begin{pmatrix} x_i \\ x_i' \end{pmatrix}$

(similar treatment for the other components of the macroparticle y_i, z_i).

  - Extension of the code: allow several bunches + MPI parallelization over the bunches (extensive use of EPFL clusters).



Parallelization quite efficient because each bunch can be treated almost independently (exchange only slices parameters between processors).

HEADTAIL Multibunch: wake field kick computation

HEADTAIL Multibunch: wake field kick computation

- Wake field kick obtained from

$$\Delta x_i' = \frac{e^2}{m_0 \gamma v^2} \sum_{z_S > z_{S_i}} n_S W_x \left(z_S - z_{S_i}, x_S, y_S, x_{S_i}, y_{S_i} \right),$$

with $W_x(z, x_S, y_S, x_{S_i}, y_{S_i}) = W_x^{dip}(z) x_S + W_{xy}^{dip}(z) y_S + W_x^{quad}(z) x_{S_i} + W_{xy}^{quad}(z) y_{S_i},$

n_S , x_S , y_S , z_S : number of particles and position of the slice *S*.

HEADTAIL Multibunch: wake field kick computation

- Wake field kick obtained from

$$\Delta x'_i = \frac{e^2}{m_0 \gamma v^2} \sum_{z_S > z_{S_i}} n_S W_x \left(z_S - z_{S_i}, x_S, y_S, x_{S_i}, y_{S_i} \right),$$

with $W_x(z, x_S, y_S, x_{S_i}, y_{S_i}) = W_x^{dip}(z) x_S + W_{xy}^{dip}(z) y_S + W_x^{quad}(z) x_{S_i} + W_{xy}^{quad}(z) y_{S_i},$

n_S , x_S , y_S , z_S : number of particles and position of the slice *S*.

- If sum runs over ALL the slices before the slice S_i of the macroparticle *i* → would take a VERY long time with many bunches / “wake memory” turns.

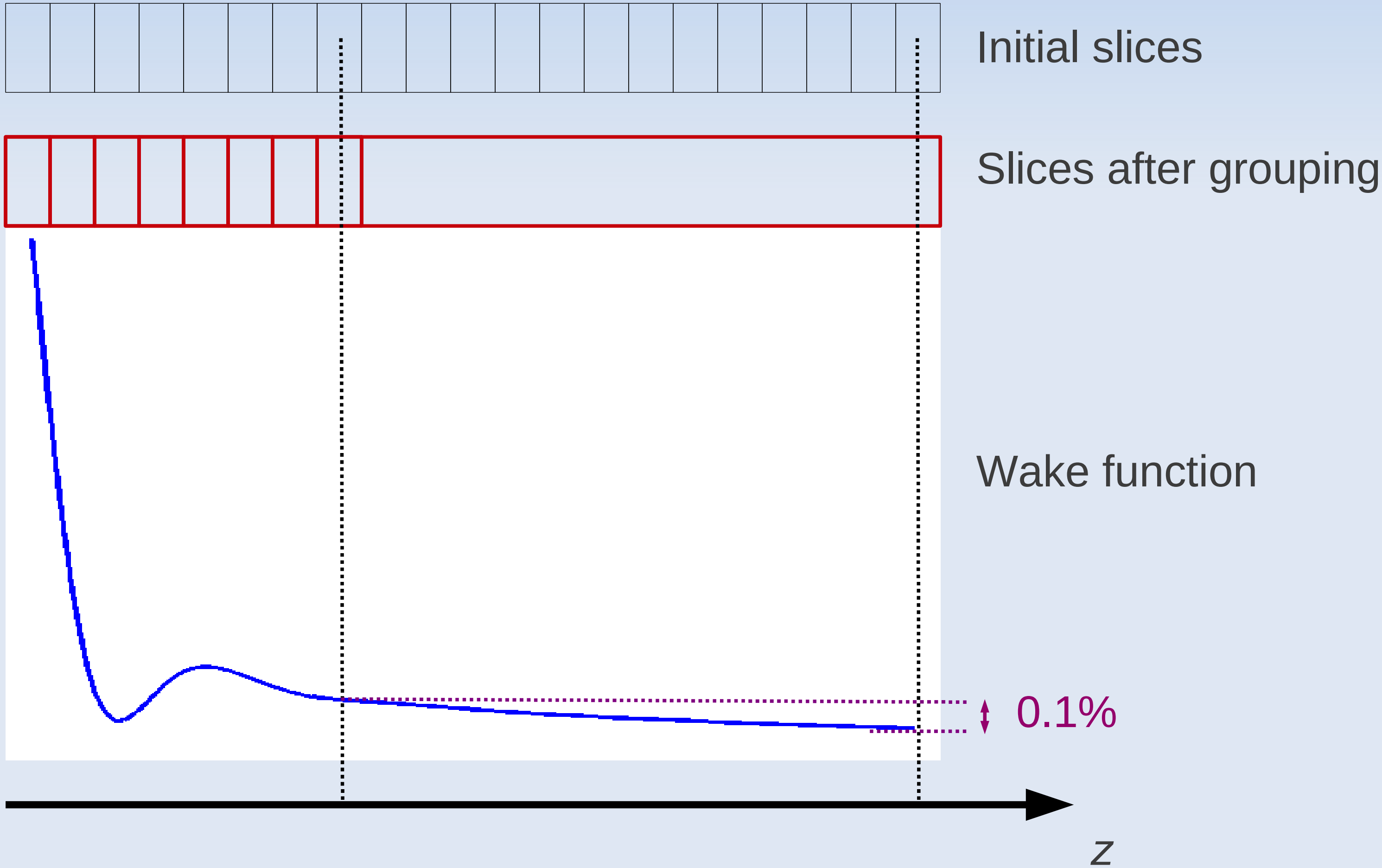
HEADTAIL Multibunch: wake field kick computation

HEADTAIL Multibunch: wake field kick computation

- Approximation to speed-up: we group together slices, bunches and even complete trains for the wake field kick computation.
- Criterion for grouping is that the wake function should *not change too much* for the group:

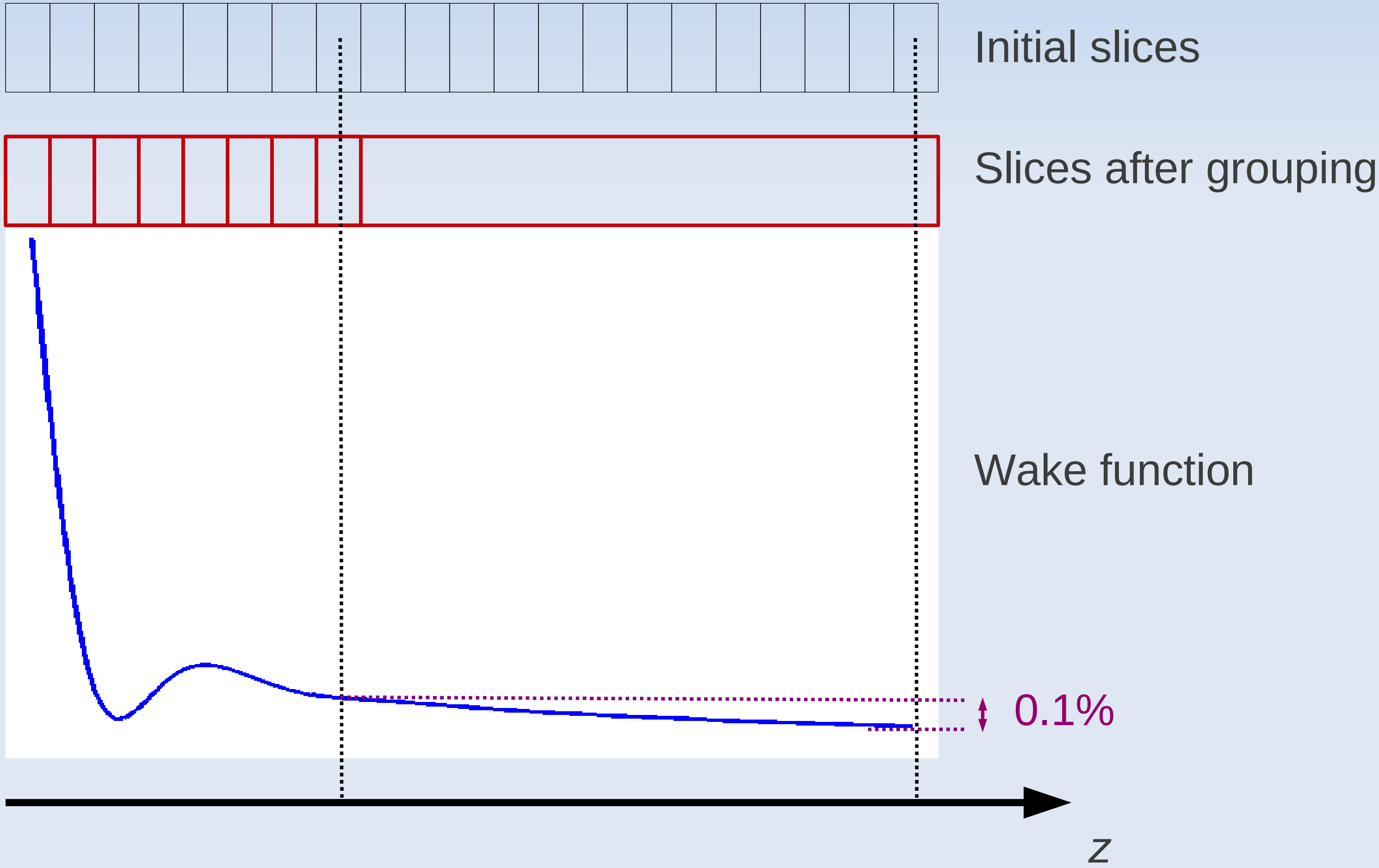
HEADTAIL Multibunch: wake field kick computation

- Approximation to speed-up: we group together slices, bunches and even complete trains for the wake field kick computation.
- Criterion for grouping is that the wake function should *not change too much* for the group:



HEADTAIL Multibunch: wake field kick computation

- Approximation to speed-up: we group together slices, bunches and even complete trains for the wake field kick computation.
- Criterion for grouping is that the wake function should *not change too much* for the group:

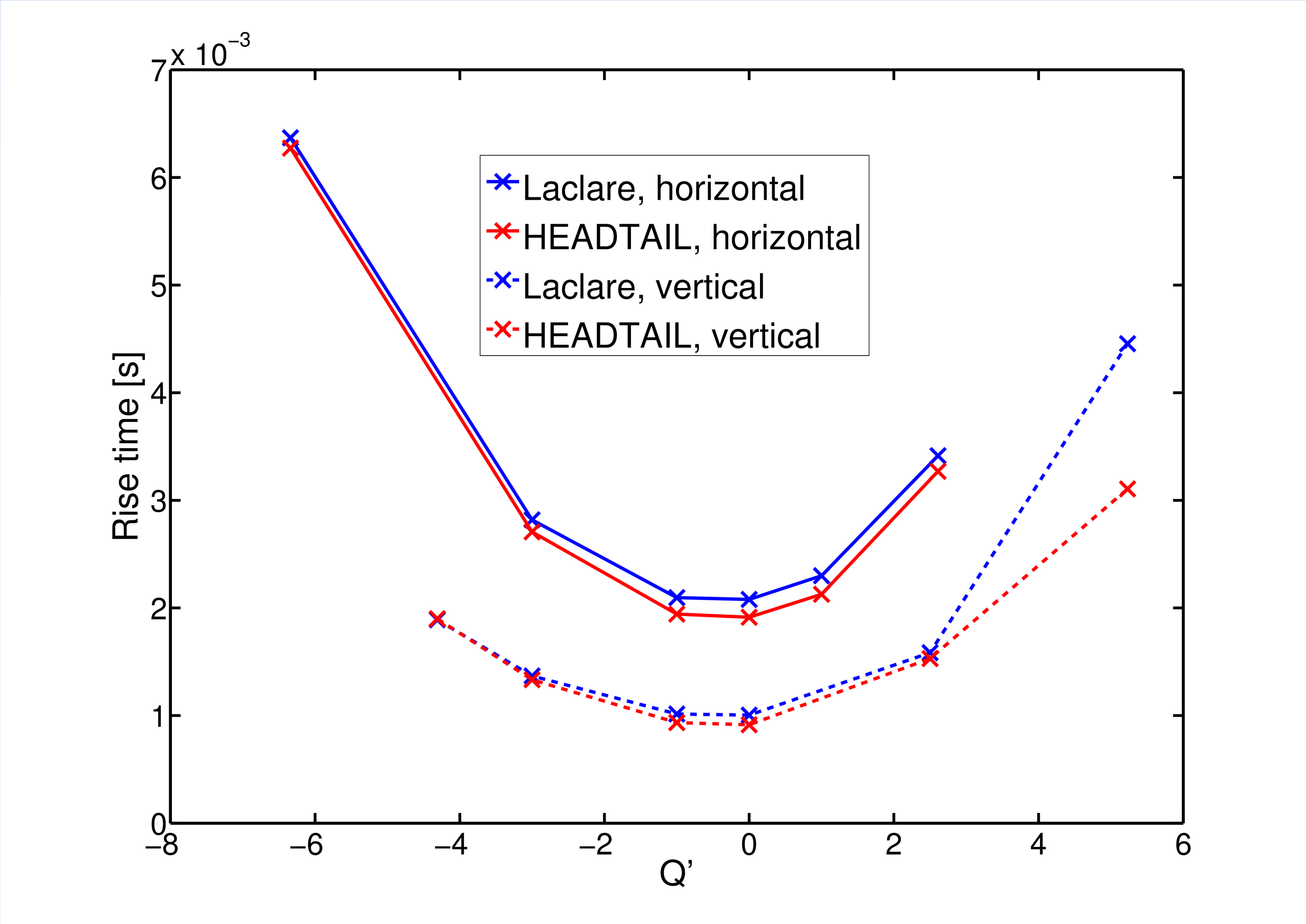


Example: for 30 turns of wake memory, 50 slices and 48 bunches → gain more than a factor 25 in computing time, with less than 0.1% impact on coherent tune shifts and rise times.

Benchmarking HEADTAIL with theory

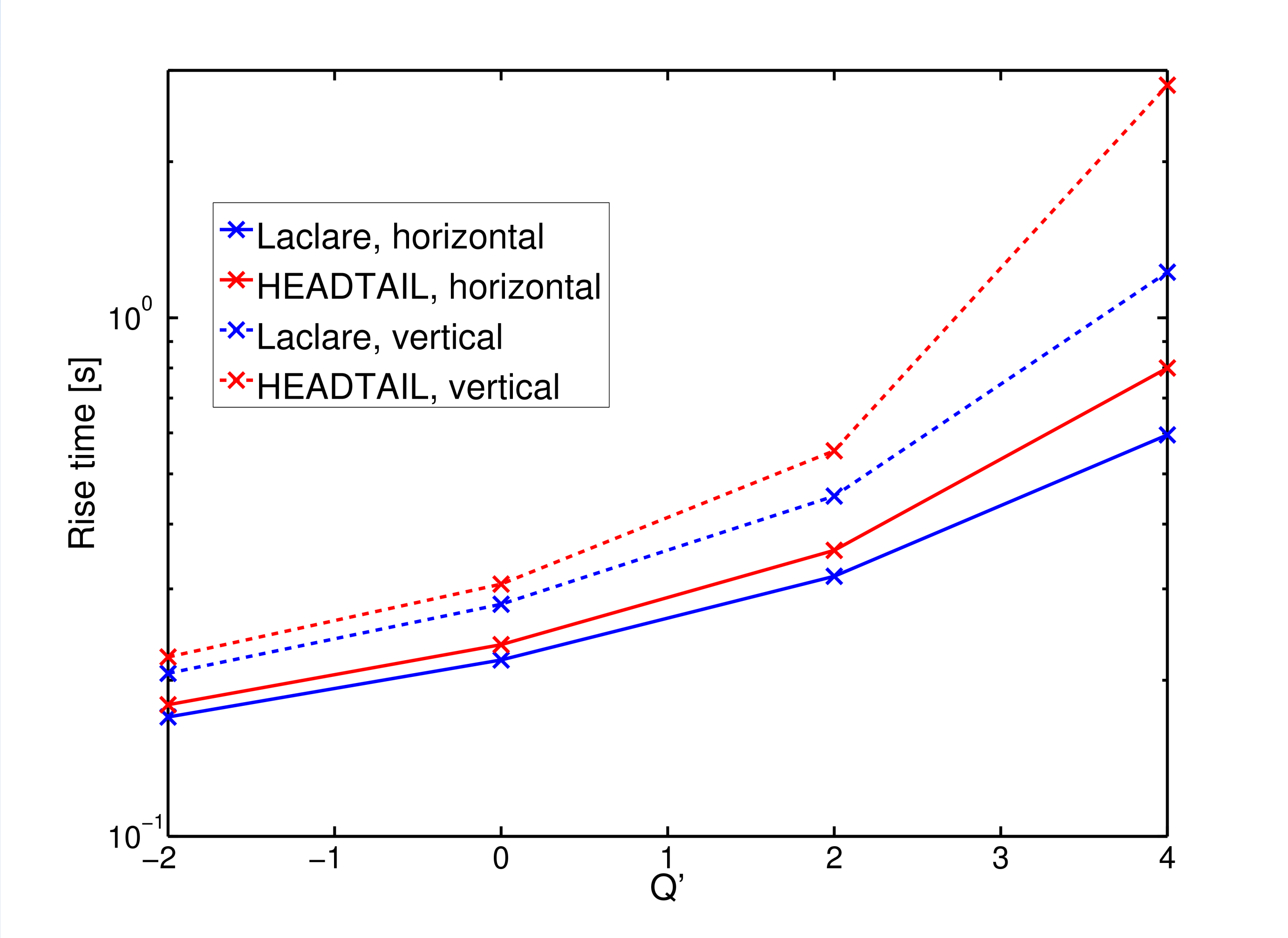
- Benchmark HEADTAIL with respect to Laclare's theory, in simplified cases (dipolar impedances & equidistant bunches):

➢ SPS beam pipe wall impedance,
➢ 924 bunches,
➢ 250000 macroparticles and 50 slices per bunch,
➢ wake memory: 19 turns,
➢ tracking 5000 turns,
➢ 16 processors used for ~3 days.



Benchmarking HEADTAIL with theory

- With LHC impedance model (wall impedance of collimators, beam screens and beam pipe + broadband model):

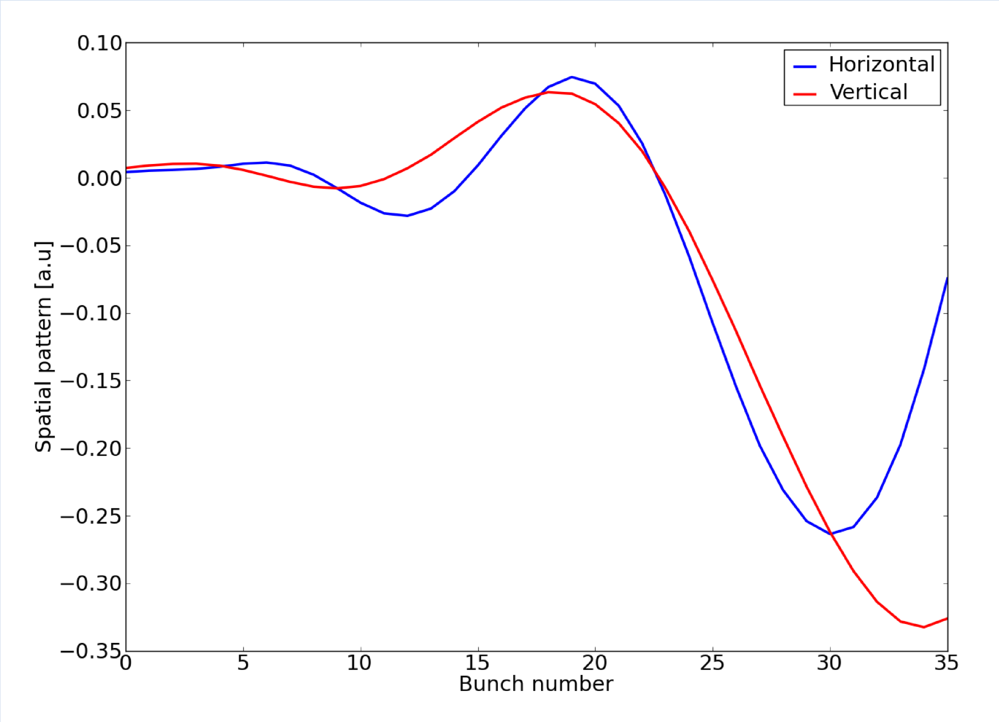


- 1782 bunches,
- 80000 macroparticles and 20 slices per bunch,
- wake memory: 19 turns,
- tracking 10000 turns,
- 48 processors used for ~3 days.

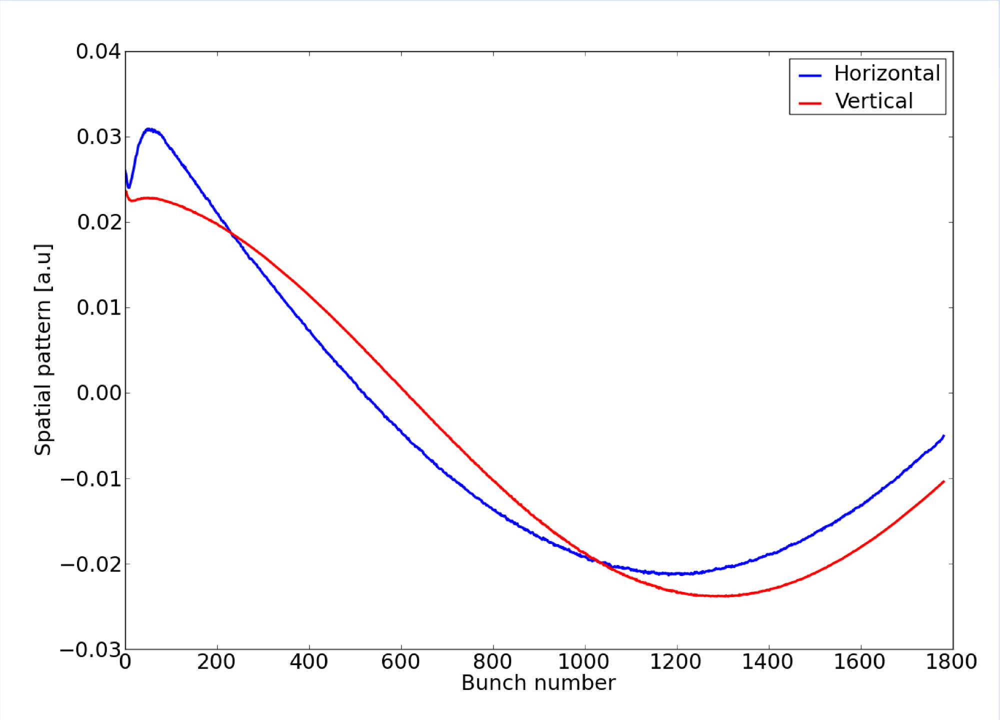
→ New HEADTAIL reliable, and also more general than available theories.

Simulations of LHC coupled-bunch instabilities: effect of the number of bunches

- The code allows to study bunch trains with non-equidistant bunches, in particular short bunch trains.
- From Singular Value Decomposition (Onishi et al, EPAC'00) of bunch centroid data (vs bunches and turns) → spatial pattern along the bunch train:



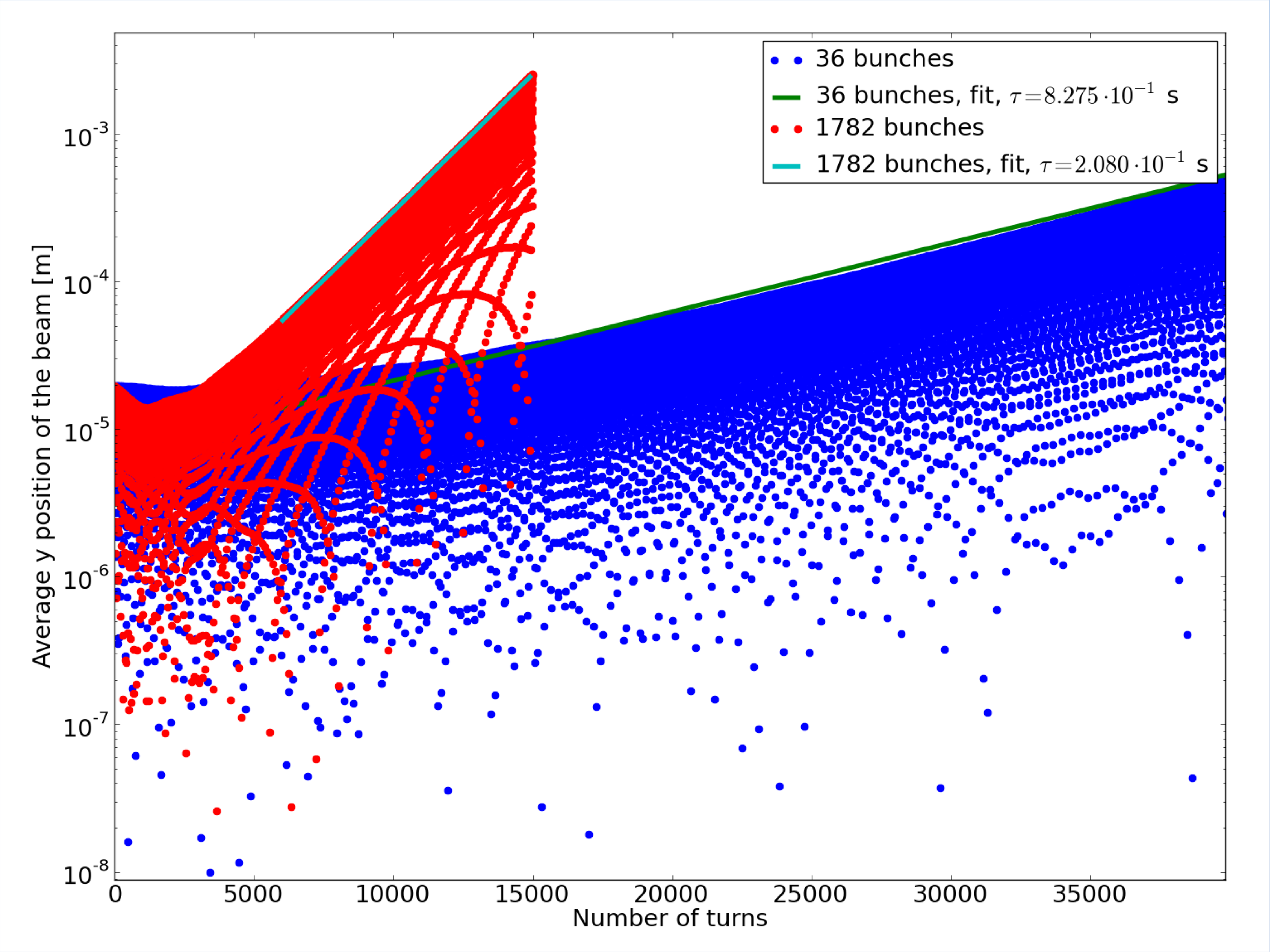
36 bunches (50 ns)



1782 bunches (50 ns)

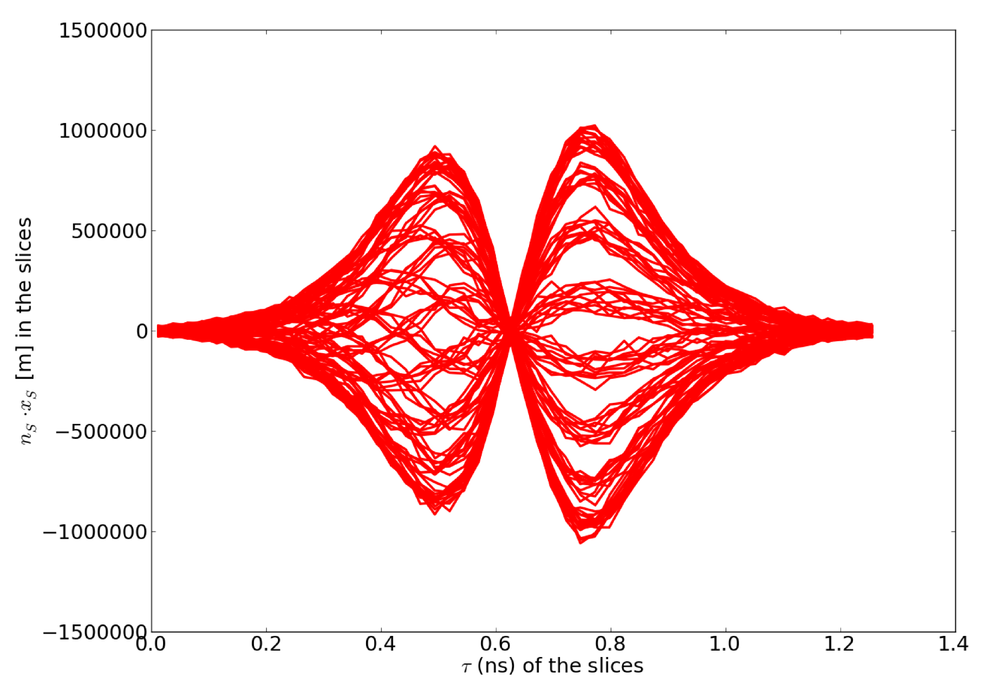
Simulations of LHC coupled-bunch instabilities: effect of the number of bunches

- A completely filled machine (1782 bunches) is at worst only 4 times more critical than a single bunch train of 36 bunches (with the same bunch spacing – 50 ns):

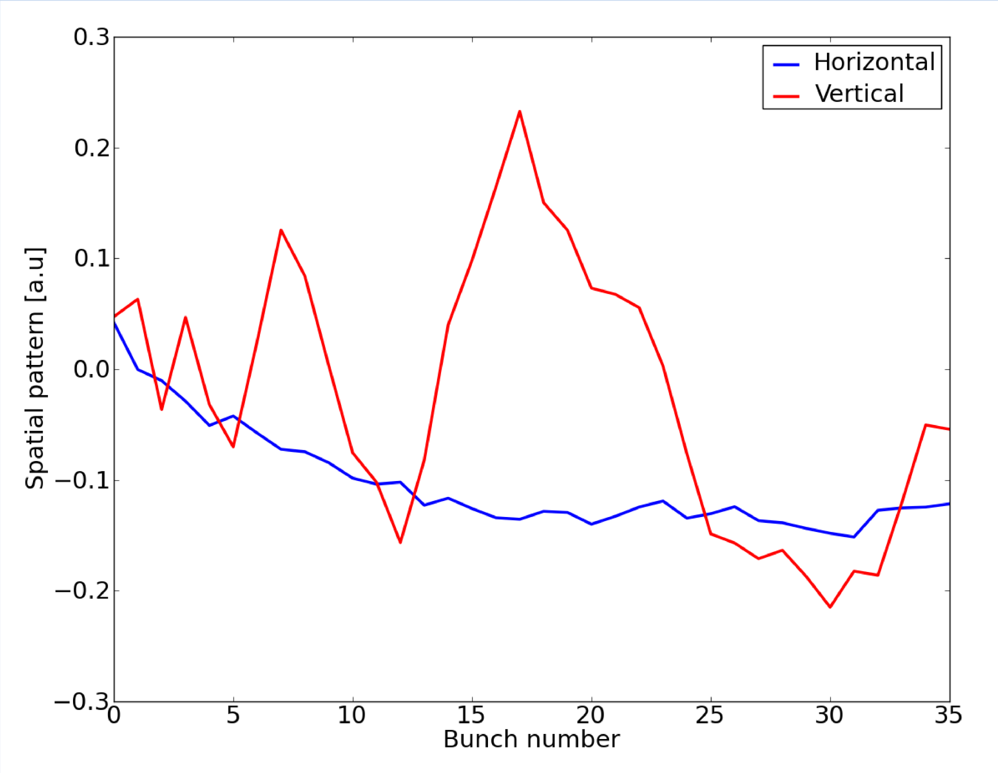


Simulations of LHC coupled-bunch instabilities with intrabunch motion

- The code allows studying coupled-bunch instabilities with non-rigid bunches: for 36 bunches (50ns spacing), with high intensity and high chromaticity:



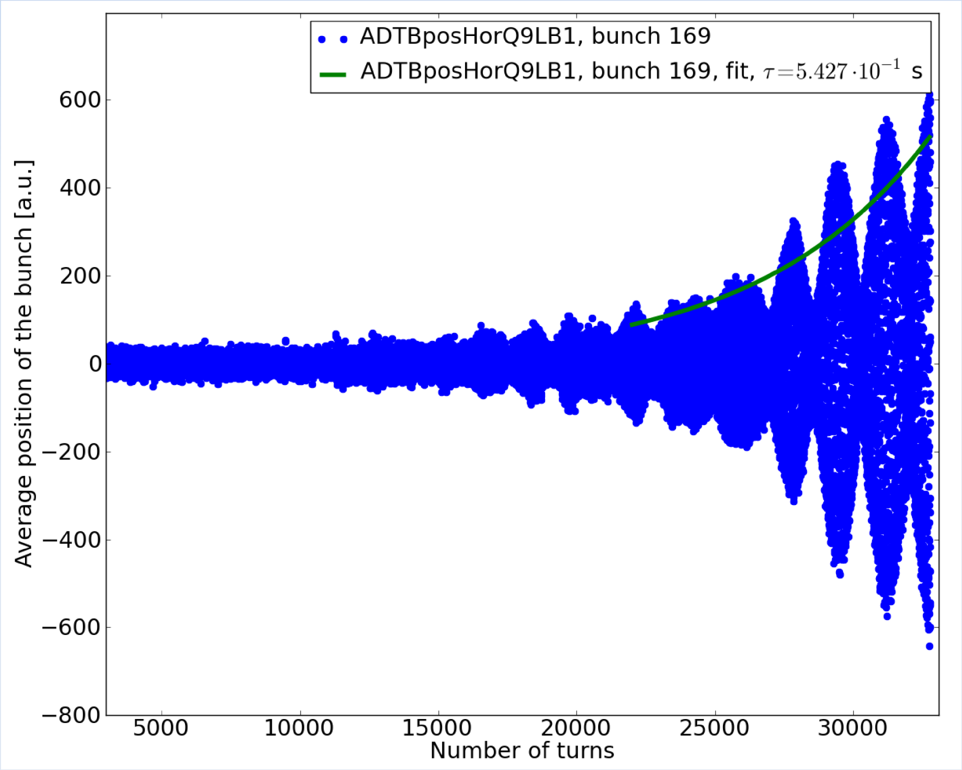
Intrabunch motion (bunch profile for subsequent turns)



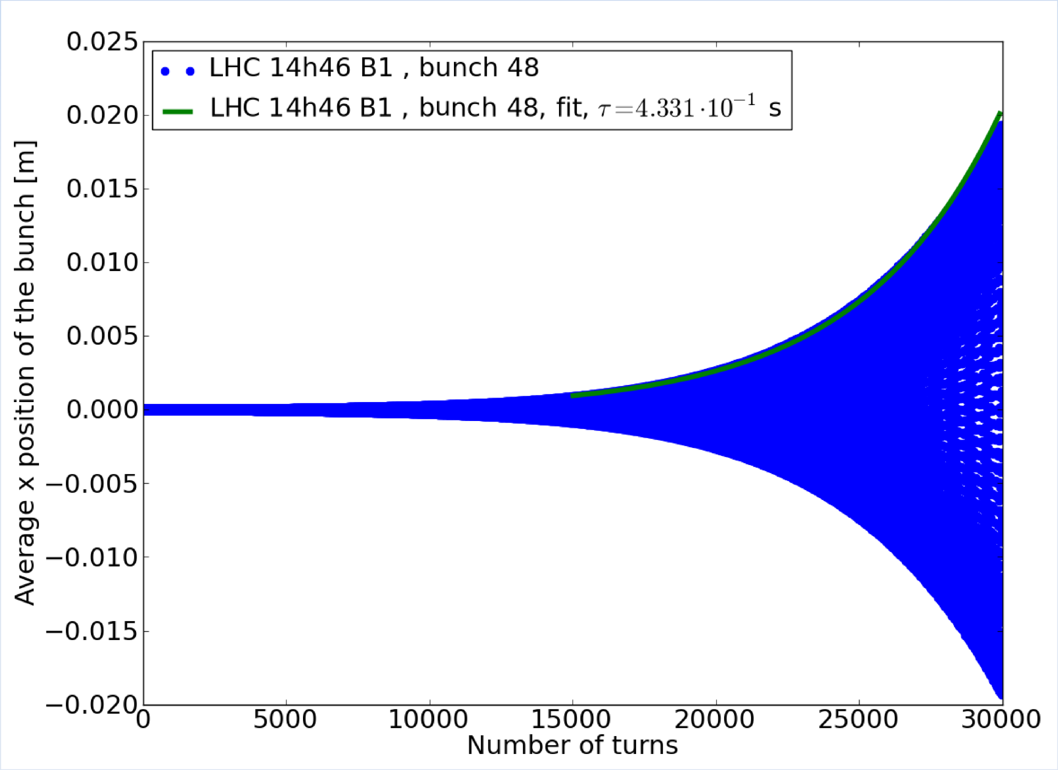
SVD spatial pattern along bunch train → coupled-bunch nature of the instability

Comparisons between simulations and beam-based impedance measurements

- At 450 GeV/*c*, 12+36 bunches, switched off feedback for 2.5 s, with $Q'_x=0.4$ → coupled-bunch instability: here for the last bunch of the train



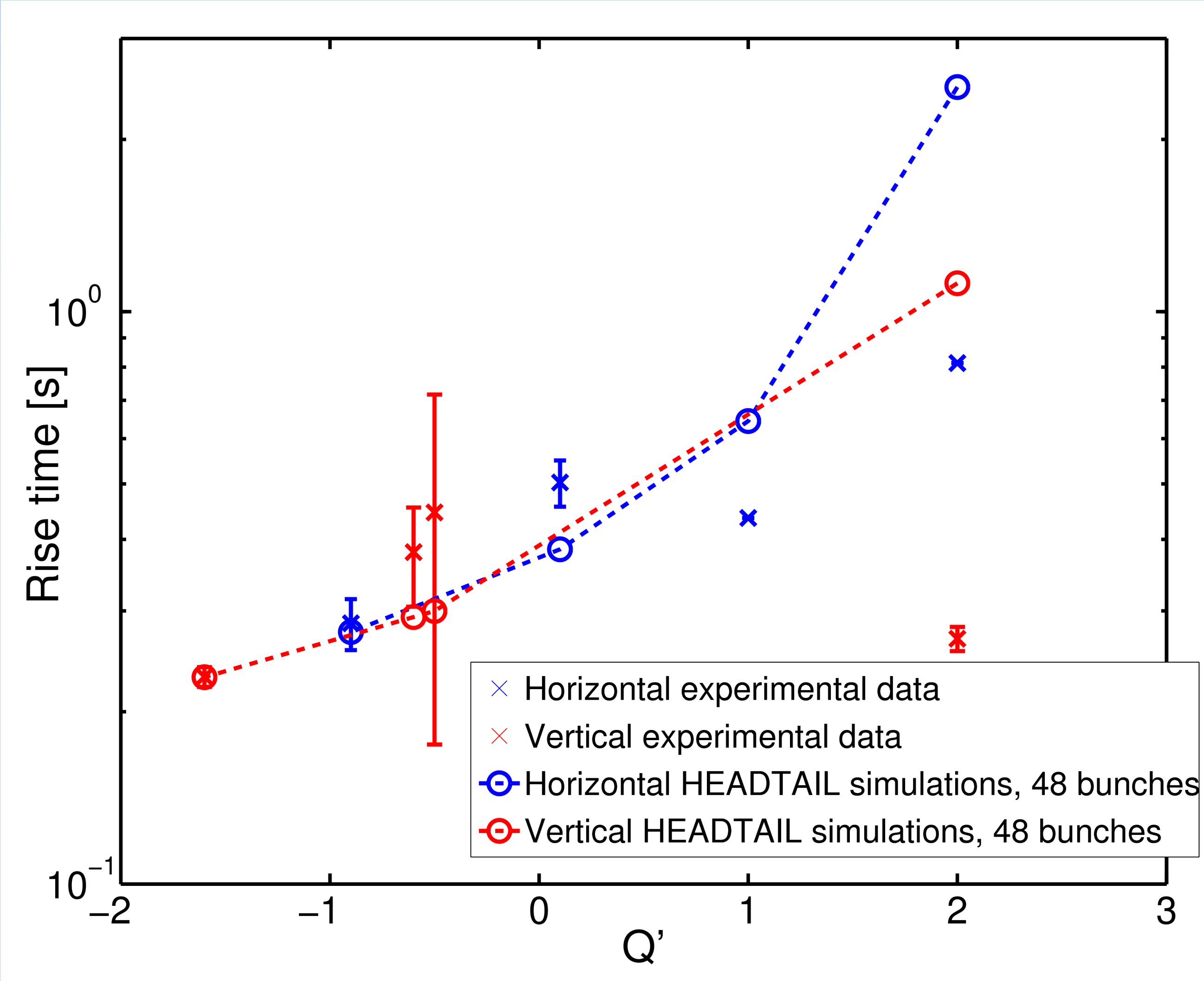
Measurement



Simulation

Comparisons between simulations and beam-based impedance measurements

- 12+36 bunches at 450GeV/*c*, coupled-bunch instability rise times measured vs. simulations (beam 2)

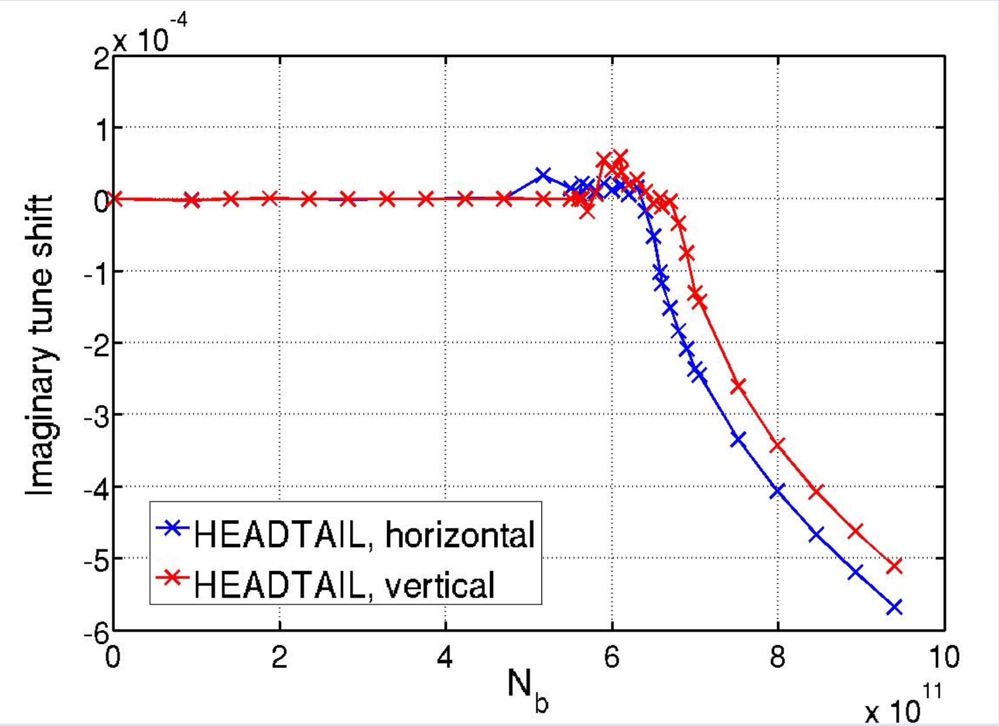


→ at this energy, measured rise times well reproduced by the model.

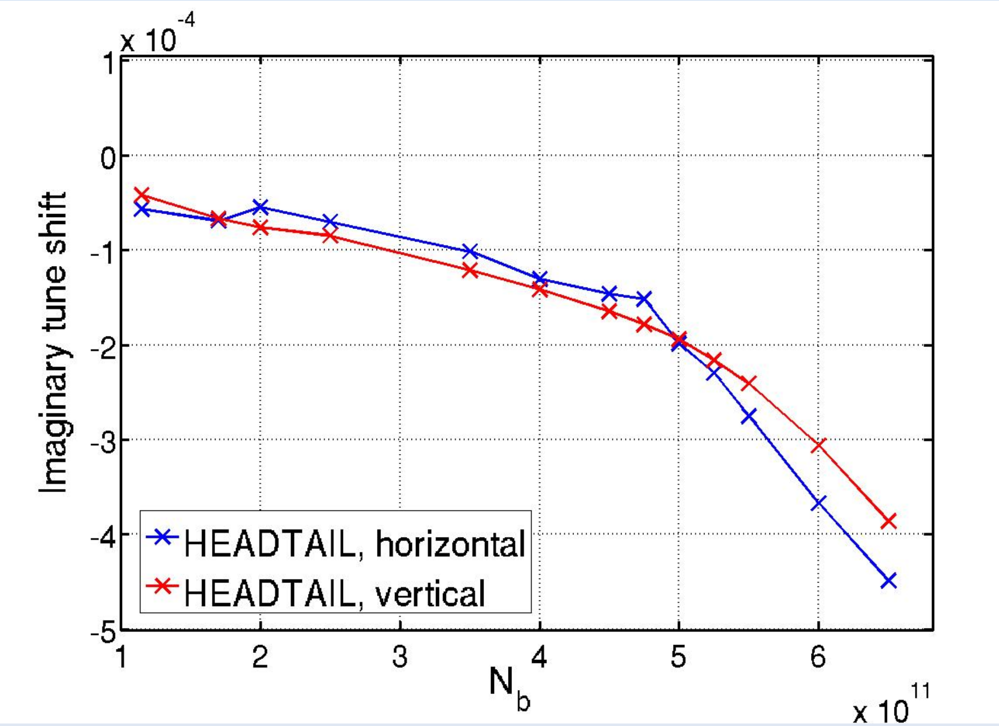
Note: at 3.5 TeV/c, measured rise times at a factor 2-3 from the model.

Predictions for the future operation of the LHC at 7 TeV/*c* : multibunch TMCI

- Transverse mode coupling instability (TMCI) intensity threshold can now be evaluated in coupled-bunch regime: at 7 TeV/c (50ns)



Single-bunch
Threshold ~ 6.4 10^{11} protons/bunch



Coupled-bunch (1404 bunches)
Threshold ~ 5 10^{11} protons/bunch

⇒ Coupled-bunch TMCI around 20% more critical than single-bunch one.

Summary

- Wake functions with both short range and long range features obtained thanks to a new algorithm to compute Fourier integrals of analytical functions → faster and more accurate than FFT.
- HEADTAIL is now able to deal with many bunches (parallelized) with all previous single-bunch features (synchrotron motion, chromaticity, octupole detuning). Benchmarked with theory → good agreement.
- Some results concerning the LHC transverse coupled-bunch instability:
  - small train of bunches vs. fully filled machine → rather small impact (at given bunch spacing),
  - coupled-bunch instabilities with intrabunch motion can be simulated,
  - comparison between measurements and simulations → reasonable agreement,
  - transverse mode coupling threshold in coupled-bunch regime → 20% lower than single-bunch threshold.

Possible future work on HEADTAIL

- Implement other sources of nonlinearities:
  - → space-charge,
  - → beam-beam force at the collision point.

- Implement a transverse damper (bunch by bunch and/or high bandwidth).

Thank you for your attention !