

Implementation of fully analytic orbit response analysis in python

Formulas used in this slides are derived from: <https://arxiv.org/abs/1711.06589v2>



S.Liuzzo, A.Franchi, June 6th 2023

The European Synchrotron

Analytic formulas for the rapid evaluation of the orbit response matrix and chromatic functions from lattice parameters in circular accelerators

Andrea Franchi and Simone Maria Liuzzo
ESRF, CS 40220, 38043 Grenoble Cedex 9, France

Zeus Martí
CELLS, 08193 Bellaterra, Spain
(Dated: September 23, 2018)

- Part 1: The implementation (12')
- Part 2: The repository (1')
- Part 3: outlook (3')

- Part 1: The implementation (12')
- Part 2: The repository (1')
- Part 3: outlook (3')

JACOBIAN OF THE ORBIT RESPONSE MATRIX

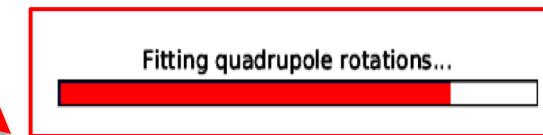
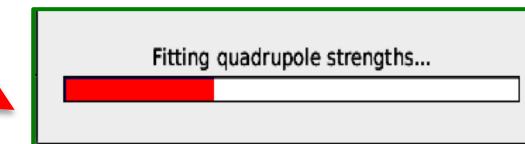
- In order to perform a linear lattice modelling and correction, the Jacobian of the ORM needs to be computed, SVD pseudo-inverted & applied to the measured ORM & dispersion.

$$\begin{pmatrix} \delta\vec{O}^{(xx)} \\ \delta\vec{O}^{(yy)} \\ \delta\vec{D}_x \end{pmatrix} = \mathbf{N} \begin{pmatrix} \delta\vec{K}_1 \\ \delta\vec{K}_0 \end{pmatrix}$$
$$\begin{pmatrix} \delta\vec{O}^{(xy)} \\ \delta\vec{O}^{(yx)} \\ \delta\vec{D}_y \end{pmatrix} = \mathbf{S} \begin{pmatrix} \vec{J}_1 \\ \vec{J}_0 \end{pmatrix} .$$

JACOBIAN OF THE ORBIT RESPONSE MATRIX

- In order to perform a linear lattice modelling and correction, the Jacobian of the ORM needs to be computed, SVD pseudo-inverted & applied to the measured ORM & dispersion.
- Numerical Jacobian: replicate in simulation the measurement: For each variation of the quadrupole strength, compute ORM from orbit calculations @ each steerer variation.
 - ✓ Pros: accurate, can be parallelized
 - ✓ Cons: time consuming, if optics or orbit unstable for quadrupole variation it can fail.

$$\begin{pmatrix} \delta\vec{O}^{(xx)} \\ \delta\vec{O}^{(yy)} \\ \delta\vec{D}_x \end{pmatrix} = \mathbf{N} \begin{pmatrix} \delta\vec{K}_1 \\ \delta\vec{K}_0 \end{pmatrix}$$
$$\begin{pmatrix} \delta\vec{O}^{(xy)} \\ \delta\vec{O}^{(yx)} \\ \delta\vec{D}_y \end{pmatrix} = \mathbf{S} \begin{pmatrix} \vec{J}_1 \\ \vec{J}_0 \end{pmatrix} .$$



JACOBIAN OF THE ORBIT RESPONSE MATRIX

- In order to perform a linear lattice modelling and correction, the Jacobian of the ORM needs to be computed, SVD pseudo-inverted & applied to the measured ORM & dispersion.
- Numerical Jacobian: replicate in simulation the measurement: For each variation of the quadrupole strength, compute ORM from orbit calculations @ each steerer variation.
 - ✓ Pros: accurate, can be parallelized
 - ✓ Cons: time consuming, if optics or orbit unstable for quadrupole variation it can fail.
- Pseudo-analytic Jacobian: use textbook formulas to evaluate ORM after computing Twiss parameters at each quadrupole variation.
 - ✓ Pros: faster than numerical, can be parallelized
 - ✓ Cons: no analytic formulas for off-diagonal ORM blocks (coupling), it can fail if optics or orbit unstable for quadrupole variation.

$$M_{ij} = \frac{\sqrt{\beta_j \beta_i}}{2 \sin \pi Q} \cos (|\phi_j - \phi_i| - \pi Q) .$$

JACOBIAN OF THE ORBIT RESPONSE MATRIX

- In order to perform a linear lattice modelling and correction, the Jacobian of the ORM needs to be computed, SVD pseudo-inverted & applied to the measured ORM & dispersion.
- Numerical Jacobian: replicate in simulation the measurement: For each variation of the quadrupole strength, compute ORM from orbit calculations @ each steerer variation.
 - ✓ Pros: accurate, can be parallelized
 - ✓ Cons: time consuming, if optics or orbit unstable for quadrupole variation it can fail.
- Pseudo-analytic Jacobian: use textbook formulas to evaluate ORM after computing Twiss parameters at each quadrupole variation.
 - ✓ Pros: faster than numerical, can be parallelized
 - ✓ Cons: no analytic formulas for off-diagonal ORM blocks (coupling), it can fail if optics or orbit unstable for quadrupole variation.
- Fully analytic Jacobian: evaluate directly the Jacobian from Twiss parameters of the initial model (ideal or from beam-based measurements)
 - ✓ Pros: coupling & dispersion included, only one computation of Twiss parameters needed, no orbit calculation needed, faster than pseudo-analytic, can be parallelized
 - ✓ Cons: tedious to code.

JACOBIAN OF THE ORBIT RESPONSE MATRIX

- In order to perform a linear lattice modelling and correction, the Jacobian of the ORM needs to be computed

- Numerical strength

$$N_{wj,m}^{(xx)} \simeq -\frac{\sqrt{\beta_{j,x}^{(mod)} \beta_{w,x}^{(mod)} \beta_{m,x}^{(mod)}}}{2 \sin(\pi Q_x^{(mod)})} \left\{ \frac{\cos(\tau_{x,wj}^{(mod)})}{4 \sin(2\pi Q_x^{(mod)})} [\cos(2\tau_{x,mj}^{(mod)}) + \cos(2\tau_{x,mw}^{(mod)})] \right. \\ \left. + \frac{\sin(\tau_{x,wj}^{(mod)})}{4 \sin(2\pi Q_x^{(mod)})} [\sin(2\tau_{x,mj}^{(mod)}) - \sin(2\tau_{x,mw}^{(mod)})] \right. \\ \left. + \frac{1}{2} \sin(\tau_{x,wj}^{(mod)}) [\Pi(m,j) - \Pi(m,w) + \Pi(j,w)] + \frac{\cos(\Delta\phi_{x,wj}^{(mod)})}{4 \sin(\pi Q_x^{(mod)})} \right\},$$

- Pseudo BPM
each quadrupole

w -> steerer

j -> BPM

m -> magnet

- Pros: fast

- Cons: noisy

- Fully analytic Jacobian: evaluate directly the Jacobian from beam-based measurements)

- Pros: coupling & dispersion included, only one computation of Twiss

pseudo-analytic, can be parallelized

- Cons: tedious to code.

$$\Pi(a,b) = 1 \quad \text{if } s_a < s_b, \quad \Pi(a,b) = 0 \quad \text{if } s_a \geq s_b$$

$$\tau_{z,ab} = \Delta\phi_{z,ab} - \pi Q_z, \quad z = x, y$$

JACOBIAN OF THE ORBIT RESPONSE MATRIX

- In order to perform a linear lattice modelling and correction, the Jacobian of the ORM needs to be computed, SVD pseudo-inverted & applied to the measured ORM & dispersion.
- Numerical Jacobian: replicate in simulation the strength, compute ORM from orbit calculations
 - ✓ Pros: accurate, can be parallelized
 - ✓ Cons: time consuming, if optics or orbit unstable for quadrupole variations
- Pseudo-analytic Jacobian: use textbook formulas for each quadrupole variation.
 - ✓ Pros: faster than numerical, can be parallelized
 - ✓ Cons: no analytic formulas for off-diagonal ORM blocks (coupling)
- Fully analytic Jacobian: evaluate directly the Jacobian from beam-based measurements)
 - ✓ Pros: coupling & dispersion included, only one computation, pseudo-analytic, can be parallelized
 - ✓ Cons: tedious to code.
- Observation: the accuracy of the two analytic approaches can be poor if thin-element model is used. Corrections to account for the variation of Twiss parameters across magnets have been included which reduce dramatically the errors w.r.t. the numerical version (see next slide).

$$\beta_m \rightarrow I_{\beta,m} = \frac{1}{L_m} \int_0^{L_m} \beta(s) ds ,$$
$$\beta_m \sin(2\tau_{mj}) \rightarrow I_{S,mj} = \frac{1}{L_m} \int_0^{L_m} \beta(s) \sin(2\tau_{sj}) ds$$
$$\beta_m \cos(2\tau_{mj}) \rightarrow I_{C,mj} = \frac{1}{L_m} \int_0^{L_m} \beta(s) \cos(2\tau_{sj}) ds$$
$$\sqrt{\beta_m} \sin(\tau_{mj}) \rightarrow J_{C,mj} = \frac{1}{L_m} \int_0^{L_m} \sqrt{\beta(s)} \sin(\tau_{sj}) ds$$
$$\sqrt{\beta_m} \cos(\tau_{mj}) \rightarrow J_{C,mj} = \frac{1}{L_m} \int_0^{L_m} \sqrt{\beta(s)} \cos(\tau_{sj}) ds$$

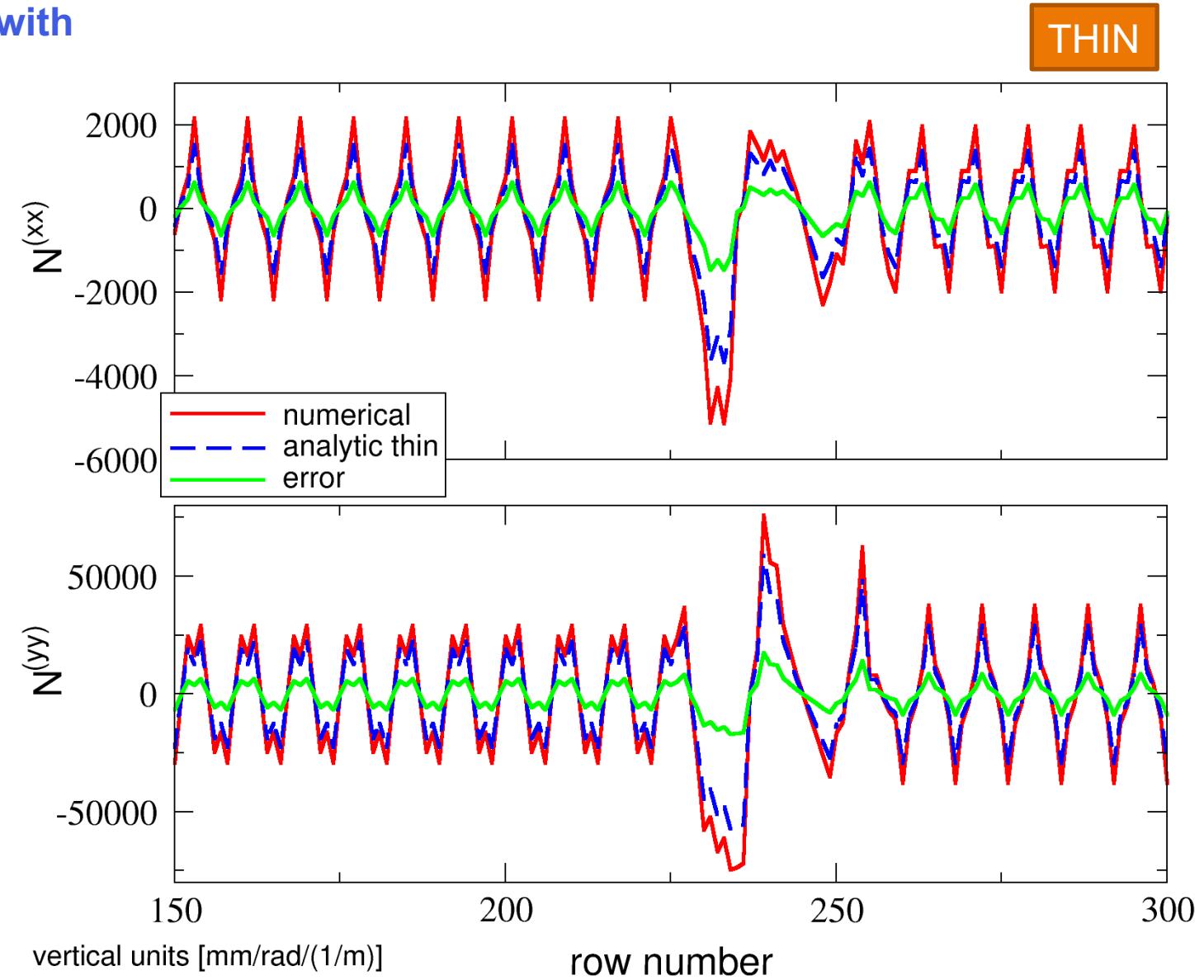
or

NUMERICAL VS ANALYTIC JACOBIAN OF THE ORM: ACCURACY

Example: FCC quadrupole ORM Jacobian with

- 1600 BPMs
- 8 steerers
- 1 quadrupole QC1L1_1

$$\begin{pmatrix} \delta\vec{O}(xx) \\ \delta\vec{O}(yy) \\ \delta\vec{D}_x \end{pmatrix} = \mathbf{N} \begin{pmatrix} \delta\vec{K}_1 \\ \delta\vec{K}_0 \end{pmatrix}$$



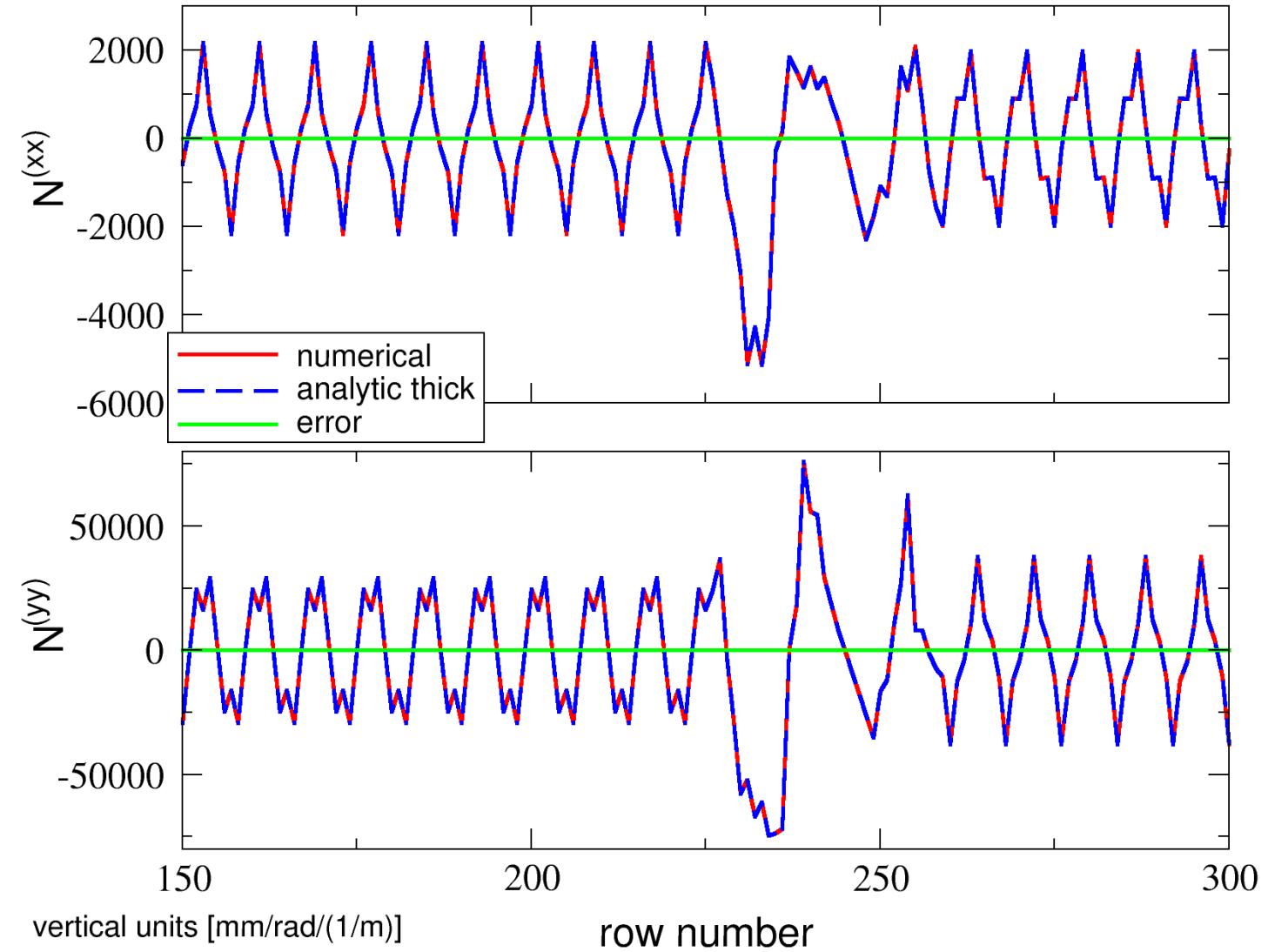
NUMERICAL VS ANALYTIC JACOBIAN OF THE ORM: ACCURACY

Example: FCC quadrupole ORM Jacobian with

THICK

- 1600 BPMs
- 8 steerers
- 1 quadrupole QC1L1_1

$$\begin{pmatrix} \delta\vec{O}(xx) \\ \delta\vec{O}(yy) \\ \delta\vec{D}_x \end{pmatrix} = \mathbf{N} \begin{pmatrix} \delta\vec{K}_1 \\ \delta\vec{K}_0 \end{pmatrix}$$



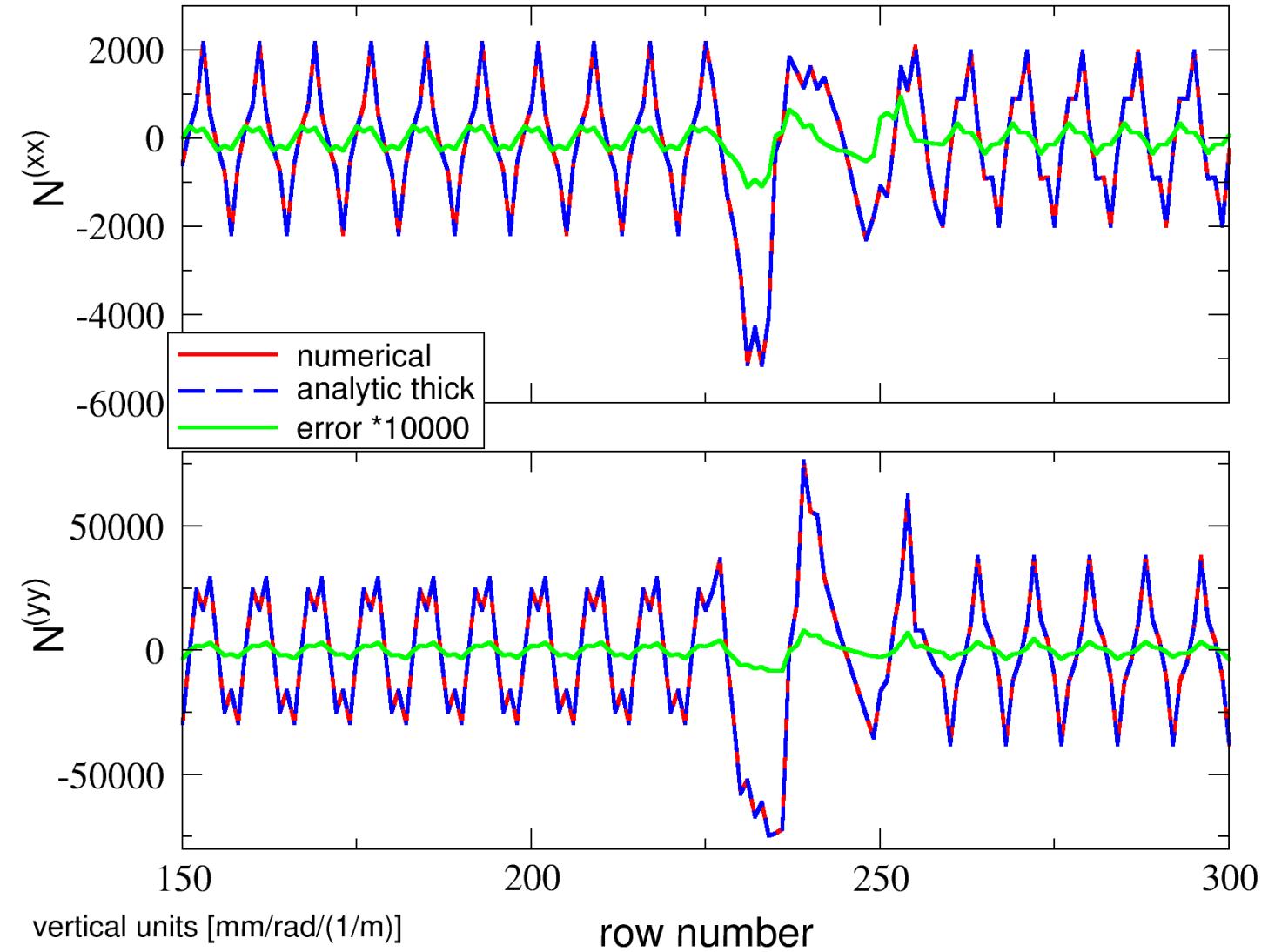
NUMERICAL VS ANALYTIC JACOBIAN OF THE ORM: ACCURACY

Example: FCC quadrupole ORM Jacobian with

THICK

- 1600 BPMs
- 8 steerers
- 1 quadrupole QC1L1_1

$$\begin{pmatrix} \delta\vec{O}(xx) \\ \delta\vec{O}(yy) \\ \delta\vec{D}_x \end{pmatrix} = \mathbf{N} \begin{pmatrix} \delta\vec{K}_1 \\ \delta\vec{K}_0 \end{pmatrix}$$

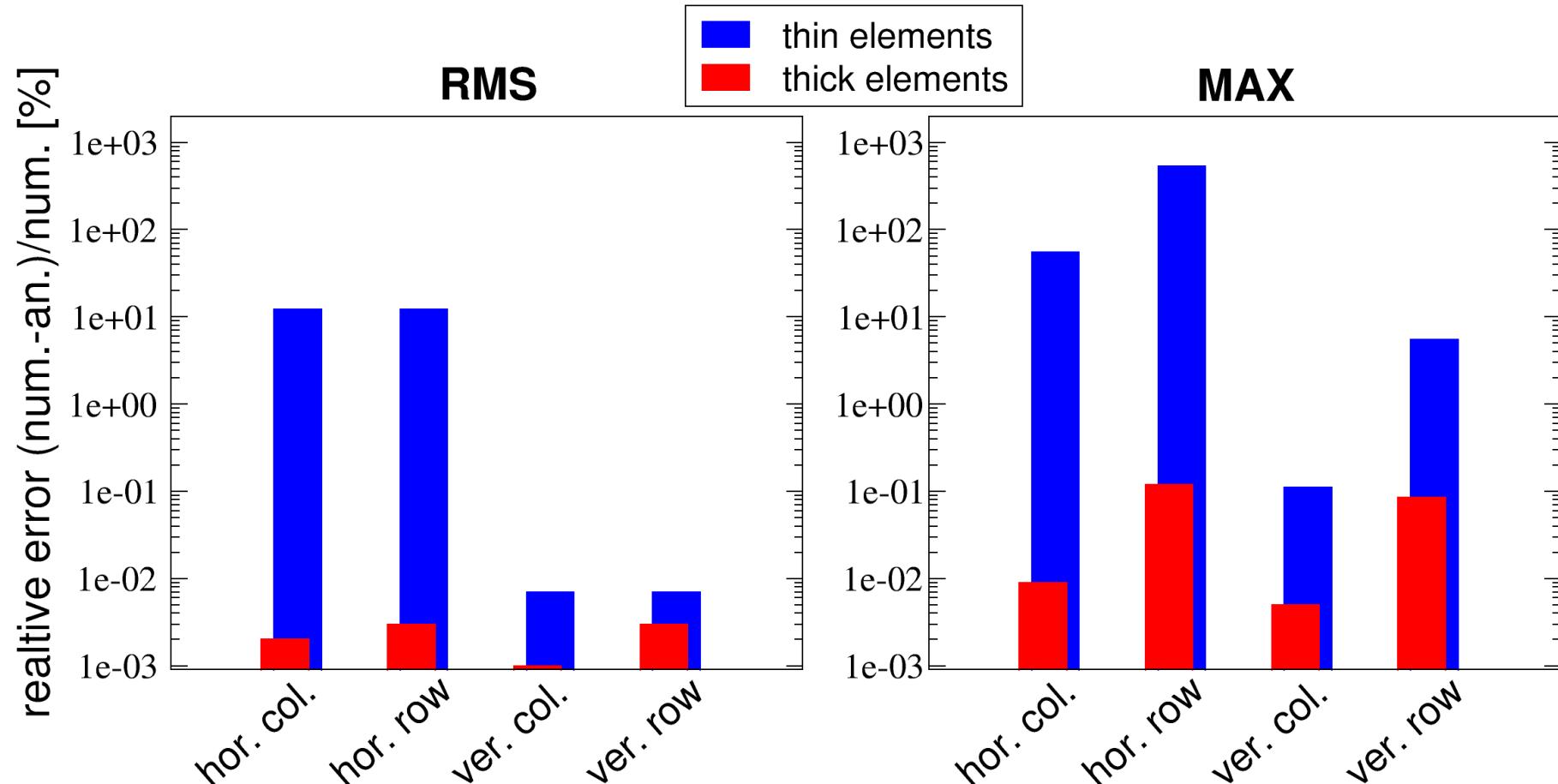


NUMERICAL VS ANALYTIC JACOBIAN OF THE ORM: ACCURACY

Example: FCC quadrupole ORM Jacobian with

- 1600 BPMs
- 8 steerers
- 1 quadrupole QC1L1_1

$$\text{ORM} = \begin{pmatrix} \mathbf{O}^{(xx)} & \mathbf{O}^{(xy)} \\ \mathbf{O}^{(yx)} & \mathbf{O}^{(yy)} \end{pmatrix}$$



- RMS & MAX error computed over all columns & rows of the diagonal ORM blocks $\mathbf{O}^{(xx)}$ & $\mathbf{O}^{(yy)}$.

Example: FCC quadrupole ORM Jacobian N (diagonal blocks only) with

- 1600 BPMs
- 8 steerers
- 360 quadrupoles parallelized over 64 cores CPUs (for both numerical and analytic tests)

Results

- Numeric: 1807.1 s [100%]
- fully analytic: 221.1 s [12%] (room for further optimization)

NUMERICAL VS ANALYTIC JACOBIAN OF THE ORM: OPTICS CORRECTION

Table 2: β -beating, dispersion and emittances after correction of 10 μm random alignment errors on dipole quadrupole and sextupole magnets for the EBS and FCC-ee lattices using analytic or numeric ORM derivative.

$\langle std \rangle_{50}$ units	$\frac{\Delta\beta_h}{\beta_{h,0}}$ %	$\frac{\Delta\beta_v}{\beta_{v,0}}$ %	$\Delta\eta_h$ mm	$\Delta\eta_v$ mm	$\Delta\epsilon_v$ pm rad
EBS					
err.	19.37	11.08	17.33	6.91	94.17
ana.	0.2	0.2	0.18	0.05	0.003
num.	0.2	0.2	0.18	0.05	0.003
FCC-ee Z					
err.	3.6	59.4	120.5	82.45	-
ana.	0.81	4.29	26.0	9.57	0.17
num.	0.82	4.30	25.98	9.64	0.18

From IPAC23 MOPL069

Table 3: β -beating, dispersion and emittances after correction of 10 μm random alignment errors on dipole quadrupole and sextupole magnets for the FCC-ee lattice using analytic ORM derivative (1856 BPMs, 18 steerers). The input lattice is tested: without radiation, with radiation and with radiation and tapering. Reference lattice is in all cases without radiation.

$\langle std \rangle_{50}$ units	$\frac{\Delta\beta_h}{\beta_{h,0}}$ %	$\frac{\Delta\beta_v}{\beta_{v,0}}$ %	$\Delta\eta_h$ mm	$\Delta\eta_v$ mm	$\Delta\epsilon_v$ pm rad
4D err	3.63	61.37	118.7	82.36	-
4D cor	0.84	4.24	25.67	9.58	0.71
6D err	3.60	59.45	120.54	82.45	-
6D cor	0.81	4.29	26.0	9.57	0.17
6D err + tapering	3.61	61.33	119.59	82.96	-
6D cor + tapering	0.82	4.22	26.03	9.65	0.18

ANALYTIC TUNE VARIATION WITH THICK QUADRUPOLES

Quad: $\Delta Q_{F1J} = 0.023\%$

EBS

error ΔQ_H (num-ana): -0.16%

error ΔQ_v (num-ana): 0.14%

THIN

$$\frac{\Delta Q}{\Delta K_j} = \frac{\beta_j}{4\pi}$$

average β function
across quadrupole

$$\frac{\Delta Q}{\Delta K_j} = \frac{\frac{1}{2} \left[\beta_j + \frac{\gamma_j}{K_j} \right] + \frac{\sin(2\sqrt{K_j}L_j)}{4\sqrt{K_j}L_j} \left[\beta_j - \frac{\gamma_j}{K_j} \right] + \frac{\alpha_j}{2K_jL_j} [\cos(2\sqrt{K_j}L_j) - 1]}{4\pi}$$

error ΔQ_H (num-ana): -0.0005 %

error ΔQ_v (num-ana): 0.0024 %

THICK

From IPAC23 MOPL069

("Thick modelling" original idea and mathematical approach : Zeus Martí , ALBA)

ANALYTIC TUNE VARIATION WITH THICK QUADRUPOLES

Quad: $\Delta Q_{FG2-1} = 0.184 \%$

FCC-ee

error ΔQ_H (num-ana): 2.2%

error ΔQ_v (num-ana): 0.38%

THIN

$$\frac{\Delta Q}{\Delta K_j} = \frac{\beta_j}{4\pi}$$

average β function
across quadrupole

$$\frac{\Delta Q}{\Delta K_j} = \frac{\frac{1}{2} \left[\beta_j + \frac{\gamma_j}{K_j} \right] + \frac{\sin(2\sqrt{K_j}L_j)}{4\sqrt{K_j}L_j} \left[\beta_j - \frac{\gamma_j}{K_j} \right] + \frac{\alpha_j}{2K_jL_j} [\cos(2\sqrt{K_j}L_j) - 1]}{4\pi}$$

error ΔQ_H (num-ana): 0.0003 %

error ΔQ_v (num-ana): -0.0008 %

THICK

From IPAC23 MOPL069

("Thick modelling" original idea and mathematical approach : Zeus Martí , ALBA)

THICK STEERERS IN CLASSIC ANALYTIC ORM FORMULA

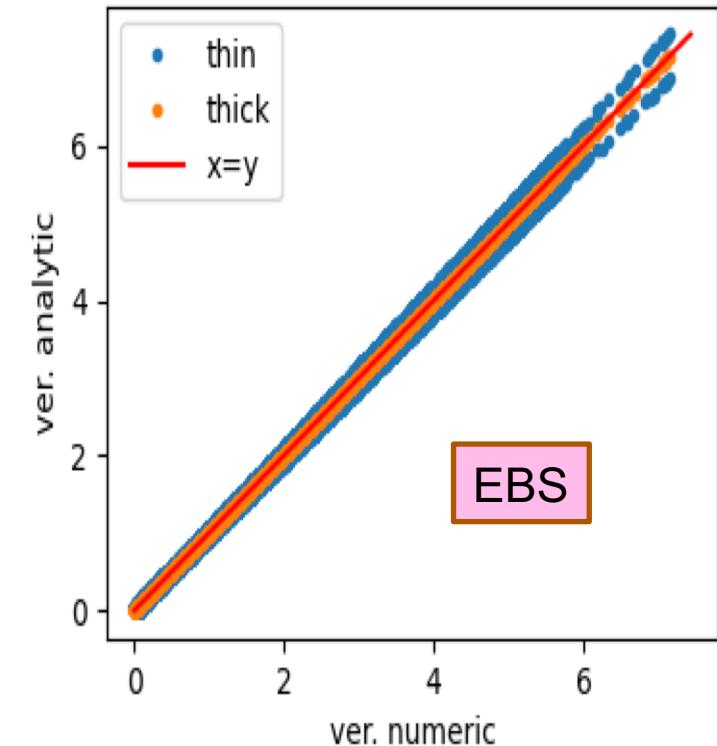
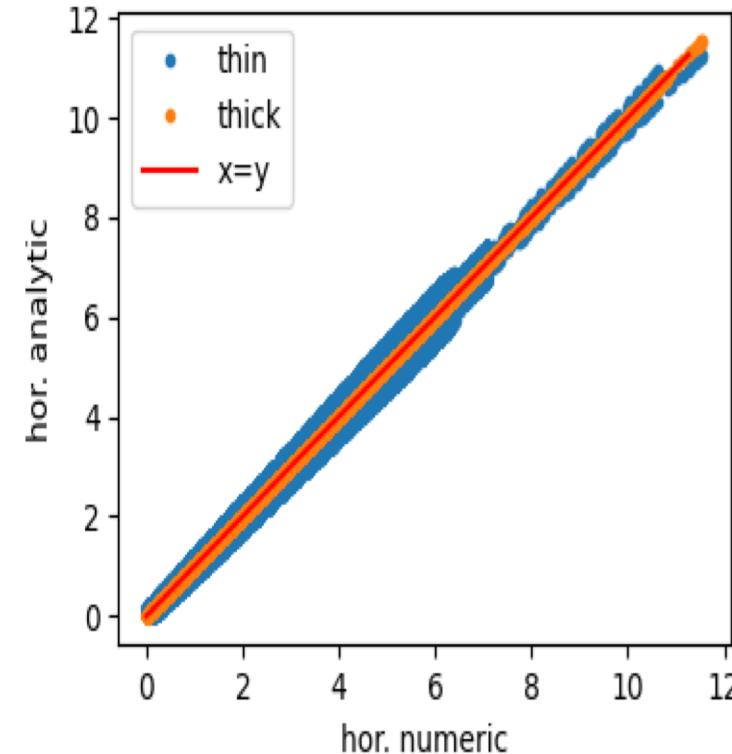
THIN steerers ORM

$$\vec{x} = M \vec{\theta} ,$$

$$M_{ij} = \frac{\sqrt{\beta_j \beta_i}}{2 \sin \pi Q} \cos (|\phi_j - \phi_i| - \pi Q) + \frac{\eta_j \eta_i}{L_0 \alpha_c}$$

THICK steerers ORM

$$M_{i,j} = \frac{\sqrt{\beta_i}}{2 \sin \pi Q} \left[\left(\sqrt{\beta_j} - \frac{\alpha_j L_j}{2 \sqrt{\beta_j}} \right) \cos (|\phi_j - \phi_i| - \pi Q) + \frac{L_j}{2 \sqrt{\beta_j}} \sin (|\phi_j - \phi_i| - \pi Q) \right] + \frac{\eta_i \eta_j}{L_0 \alpha_c}$$

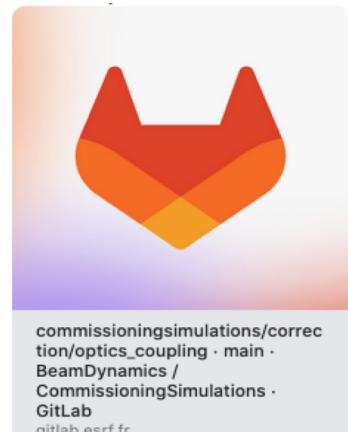


- Part 1: The implementation (12')
- **Part 2: The repository (1')**
- Part 3: outlook (3')

<https://gitlab.esrf.fr/BeamDynamics/commissioningsimulations>

Commissioning tools are still poor in terms of documentation, and debugging. The inclusion into the pyAT repository is pending such extensive validation tests.

Modules for the fully analytic ORM Jacobian can be found here:



https://gitlab.esrf.fr/BeamDynamics/commissioningsimulations/-/tree/main/commissioningsimulations/correction/optics_coupling

- Part 1: The implementation (12')
- Part 2: The repository (1')
- Part 3: outlook (3')

OUTLOOK: THICK-ELEMENT CORRECTION FOR RDTs

$$f_{jklm}(s) \propto \sum_m \beta_{m,x}^{(J+k)/2} \beta_{m,y}^{(l+m)/2} e^{i[(j-k)\Delta\phi_{w,x}^{(s)} + (l-m)\Delta\phi_{w,y}^{(s)}]}$$

$$f_{1001,j} = \frac{\sum_{m=1}^M J_{m,1} \sqrt{\beta_{m,x}\beta_{m,y}} e^{i(\Delta\phi_{x,mj} - \Delta\phi_{y,mj})}}{4 [1 - e^{2\pi i(Q_x - Q_y)}]} + O(J_1^2)$$

$$f_{1010,j} = \frac{\sum_{m=1}^M J_{m,1} \sqrt{\beta_{m,x}\beta_{m,y}} e^{i(\Delta\phi_{x,mj} + \Delta\phi_{y,mj})}}{4 [1 - e^{2\pi i(Q_x + Q_y)}]} + O(J_1^2)$$

$$f_{2000,j} = -\frac{\sum_{m=1}^M \beta_{m,x}^{(mod)} \delta K_{m,1} e^{2i\Delta\phi_{x,mj}^{(mod)}}}{1 - e^{4\pi i Q_x^{(mod)}}} + O(\delta K_1^2)$$

$$f_{0020,j} = \frac{\sum_{m=1}^M \beta_{m,y}^{(mod)} \delta K_{m,1} e^{2i\Delta\phi_{y,mj}^{(mod)}}}{1 - e^{4\pi i Q_y^{(mod)}}} + O(\delta K_1^2)$$

$$\beta_m^A \sin(\mathbf{B} \Delta\phi_{mj}) \rightarrow \frac{1}{L_m} \int_0^{L_m} \beta(s)^A \sin(\mathbf{B} \Delta\phi_s) ds$$

$$\beta_m^A \cos(\mathbf{B} \Delta\phi_{mj}) \rightarrow \frac{1}{L_m} \int_0^{L_m} \beta(s)^A \cos(\mathbf{B} \Delta\phi_s) ds$$

Thick-element corrections are being implemented to RDTs

OUTLOOK: A NEW WAY TO COMPUTE LINEAR RDTs

From analytic formulas

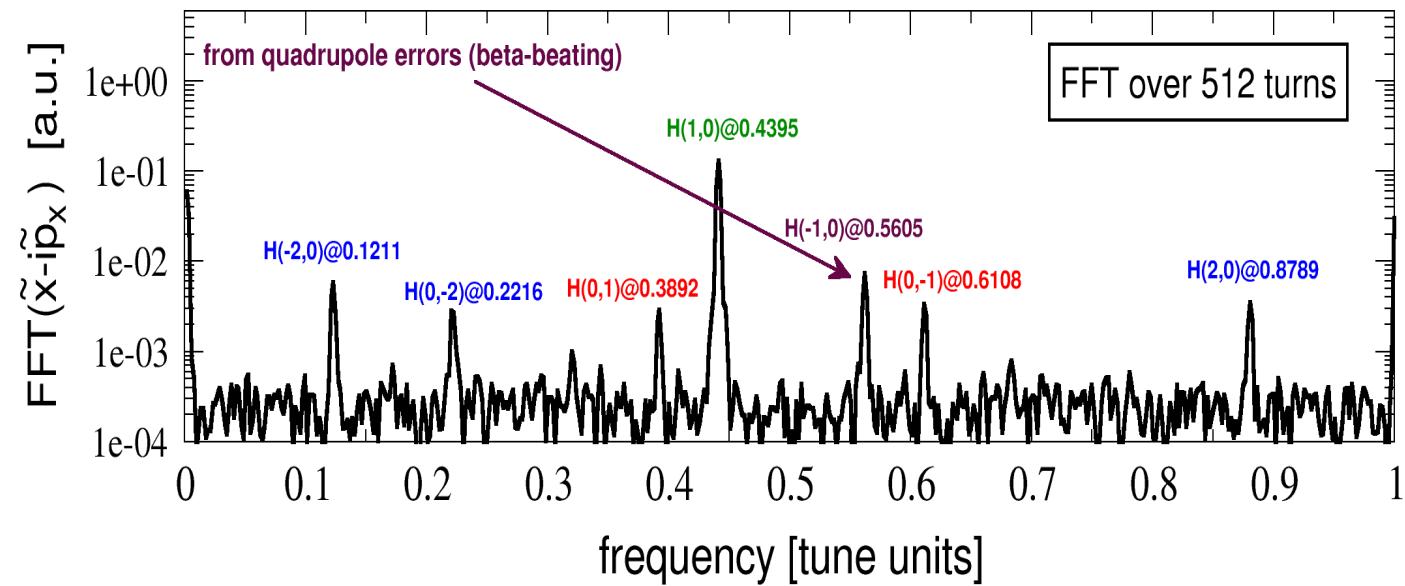
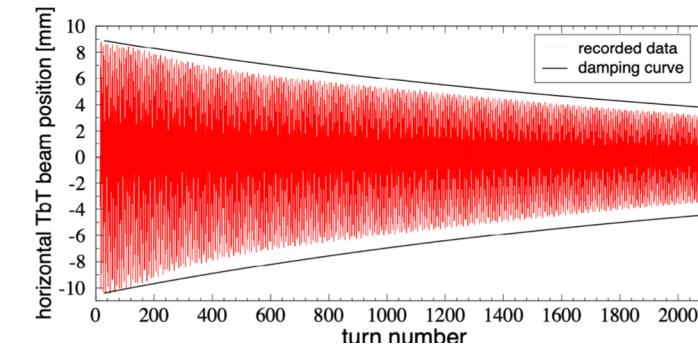
$$f_{2000,j} = -\frac{\sum_{m=1}^M \beta_{m,x}^{(mod)} \delta K_{m,1} e^{2i\Delta\phi_{x,mj}^{(mod)}}}{1 - e^{4\pi i Q_x^{(mod)}}} + O(\delta K_1^2)$$

$$f_{0020,j} = \frac{\sum_{m=1}^M \beta_{m,y}^{(mod)} \delta K_{m,1} e^{2i\Delta\phi_{y,mj}^{(mod)}}}{1 - e^{4\pi i Q_y^{(mod)}}} + O(\delta K_1^2)$$

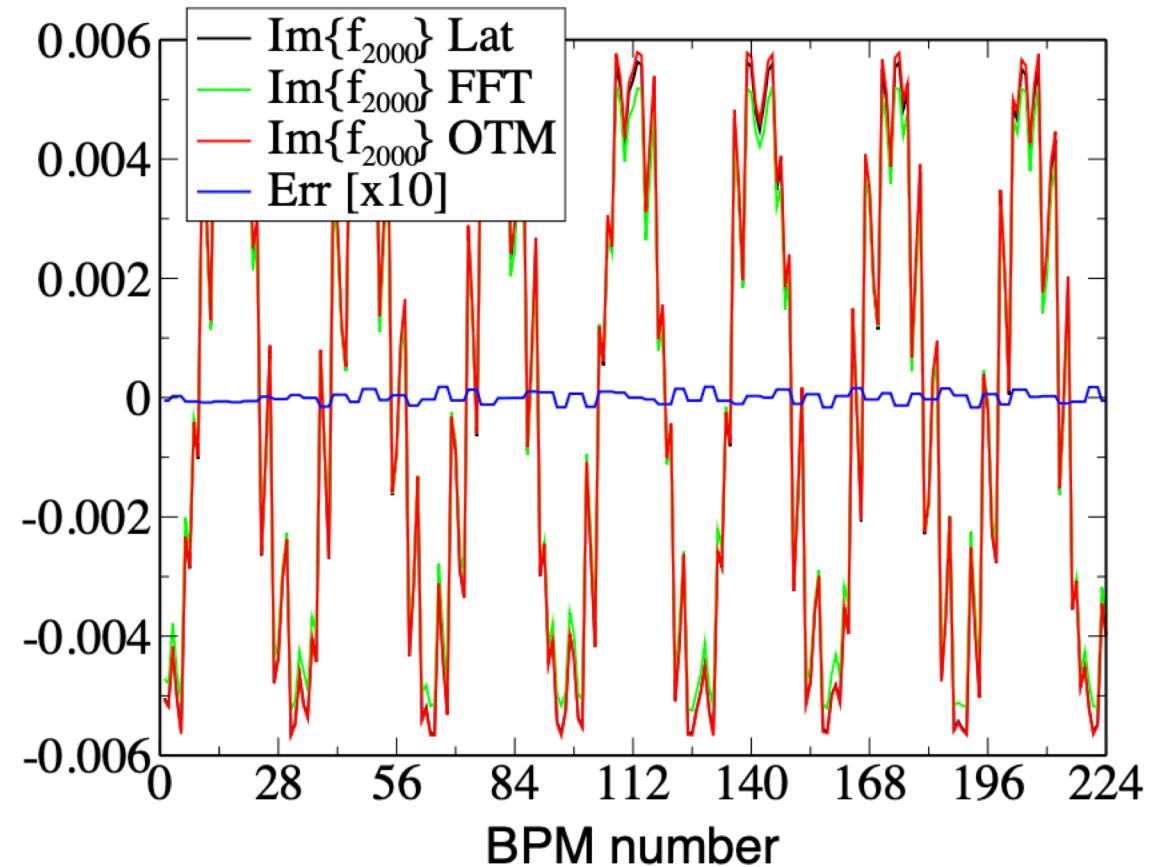
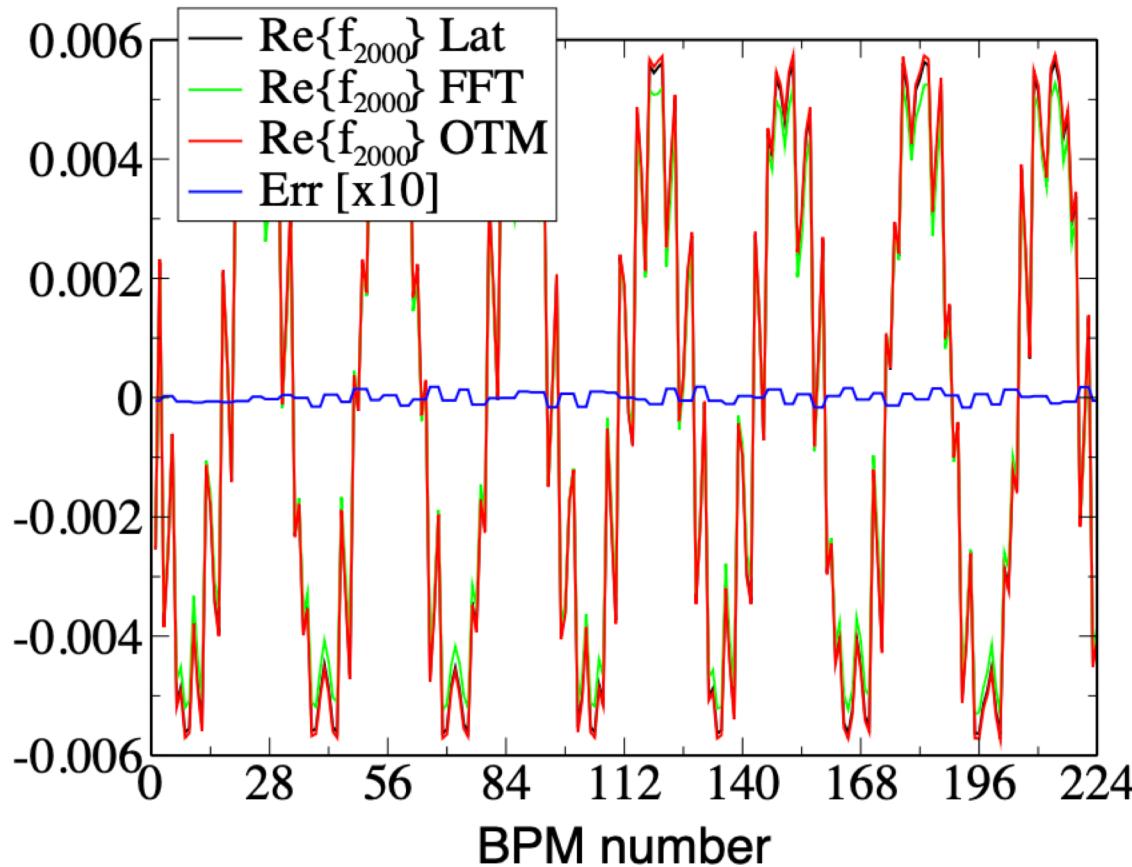
New formulas from OTM

$$\vec{X}^{(N+1)} = \mathbf{M} \vec{X}^{(N)}, \quad \vec{X} = \begin{pmatrix} x \\ p_x \\ y \\ p_y \end{pmatrix}$$

From particle tracking + harmonic analysis

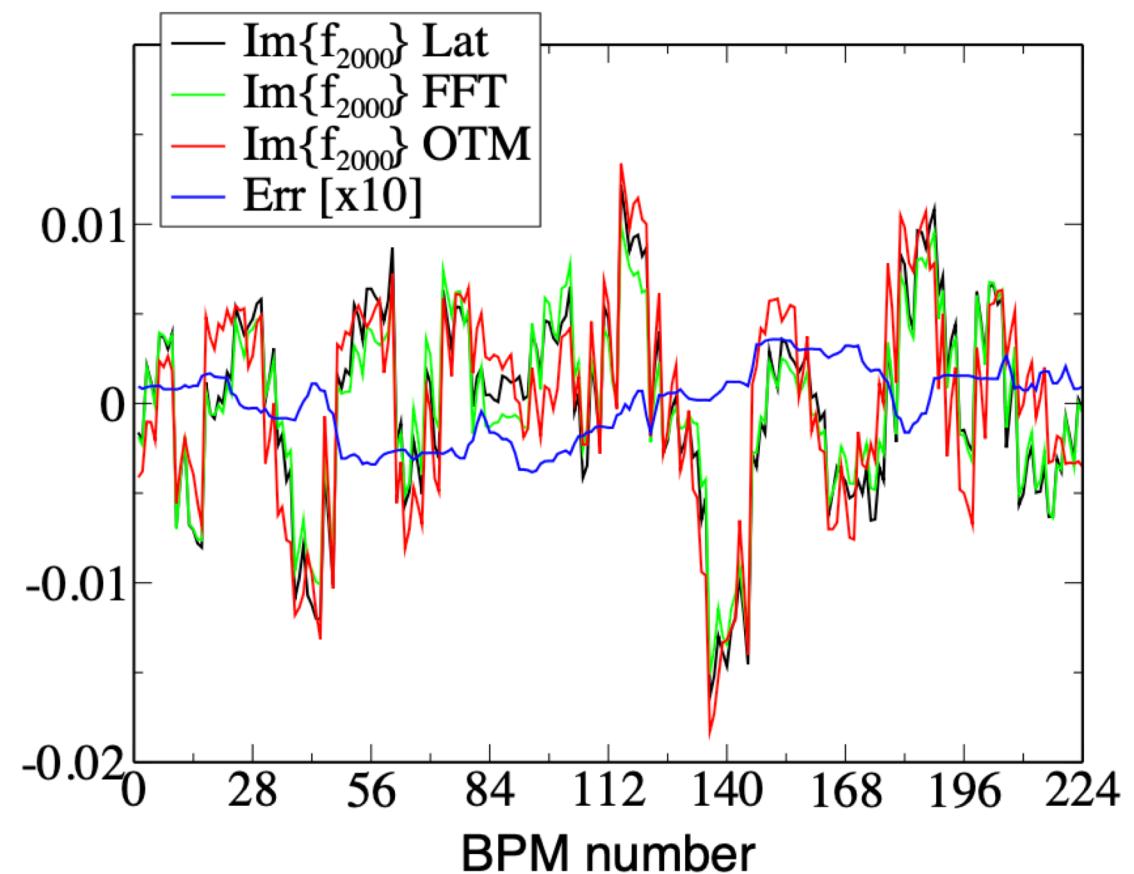
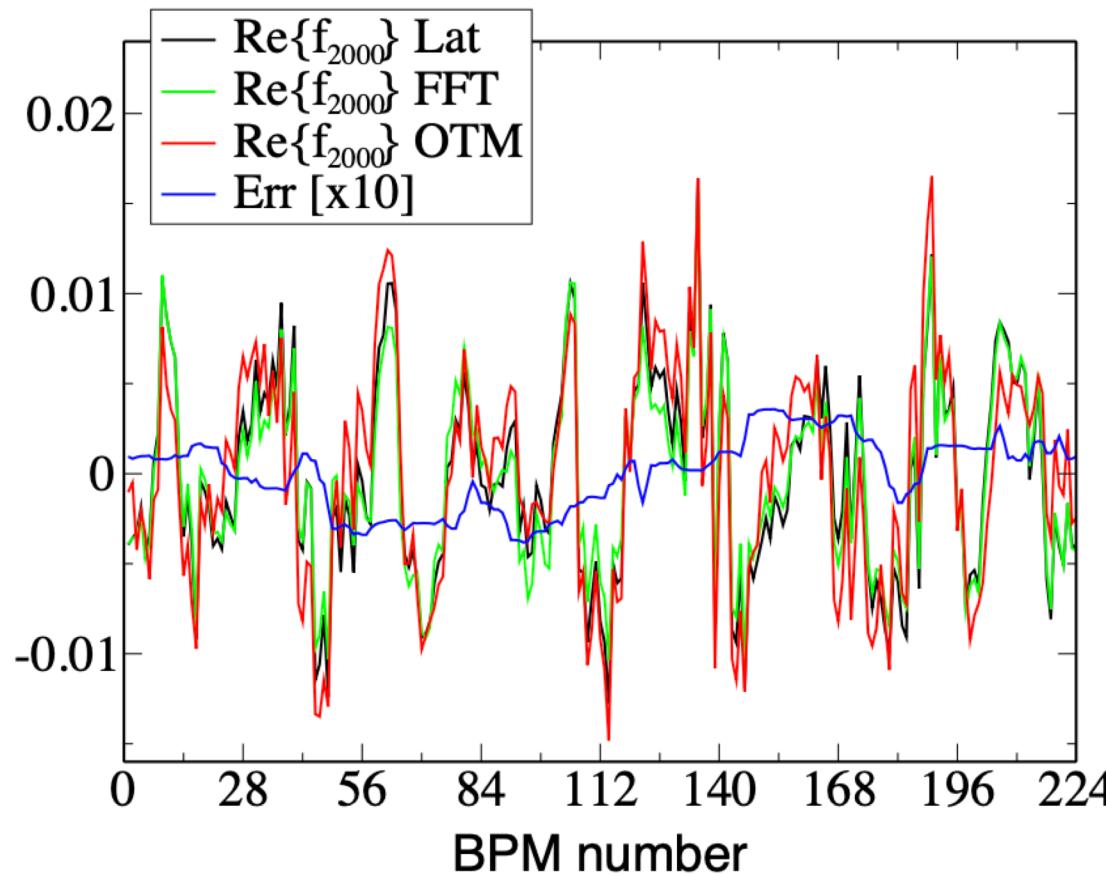


OUTLOOK: A NEW WAY TO COMPUTE LINEAR RDTS



Very accurate for single-quad error (thus ok for response matrix)

OUTLOOK: A NEW WAY TO COMPUTE LINEAR RDTS



Less accurate for distributed errors (2nd order & coupling terms)

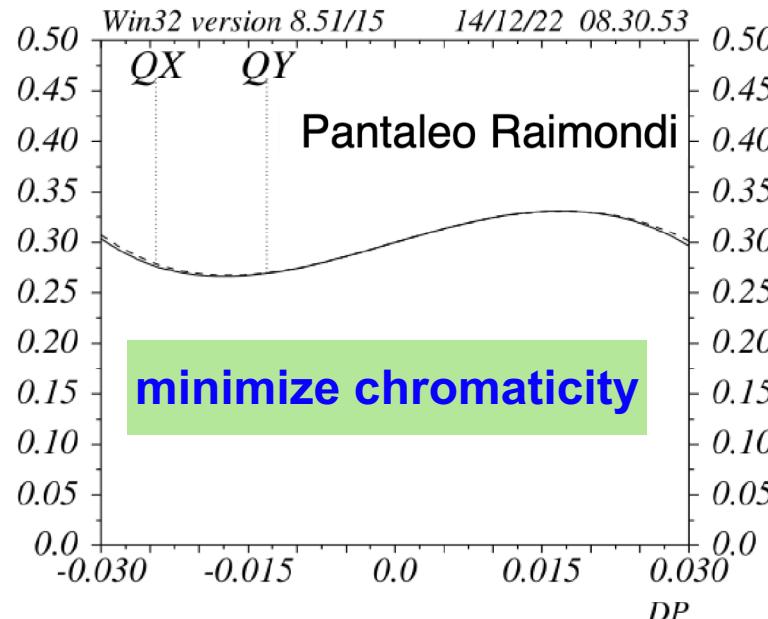
New formulas for coupling RDTs from OTM are being derived

$$\vec{X}^{(N+1)} = \mathbf{M} \vec{X}^{(N)}, \quad \vec{X} = \begin{pmatrix} x \\ p_x \\ y \\ p_y \end{pmatrix}$$

- Part 1: The implementation (12')
- Part 2: The repository (1')
- Part 3: outlook (3')

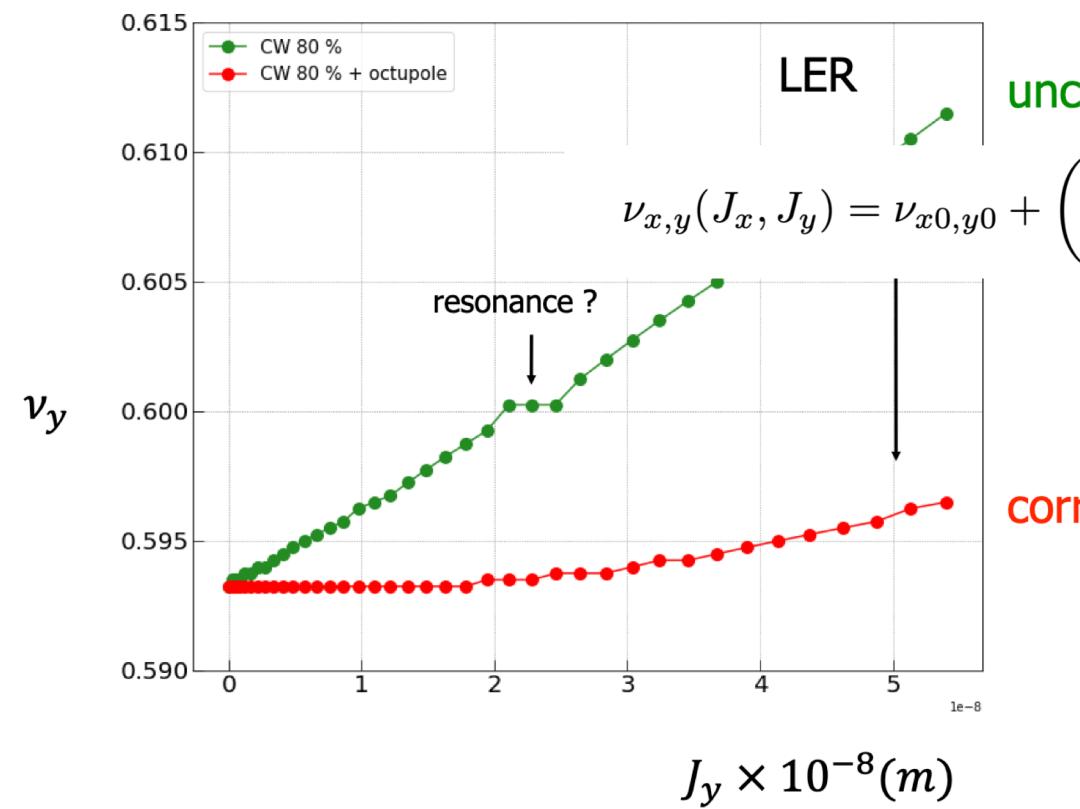
Thank you!

SOME THOUGHTS ON NON-LINEAR OPTIMIZATION



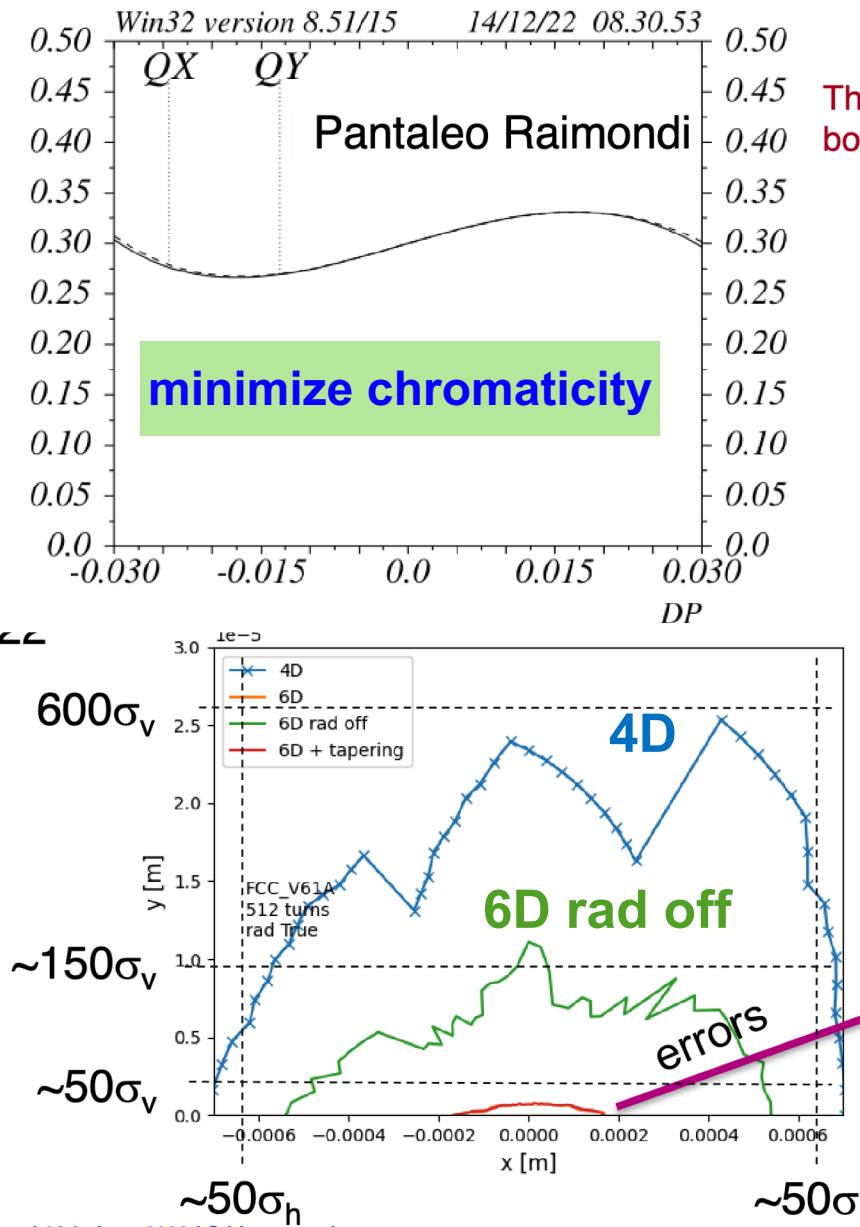
The ARC+LSS has zero second order chromaticity in both planes, only third and higher orders remain.

Y. Ohnishi / KEK



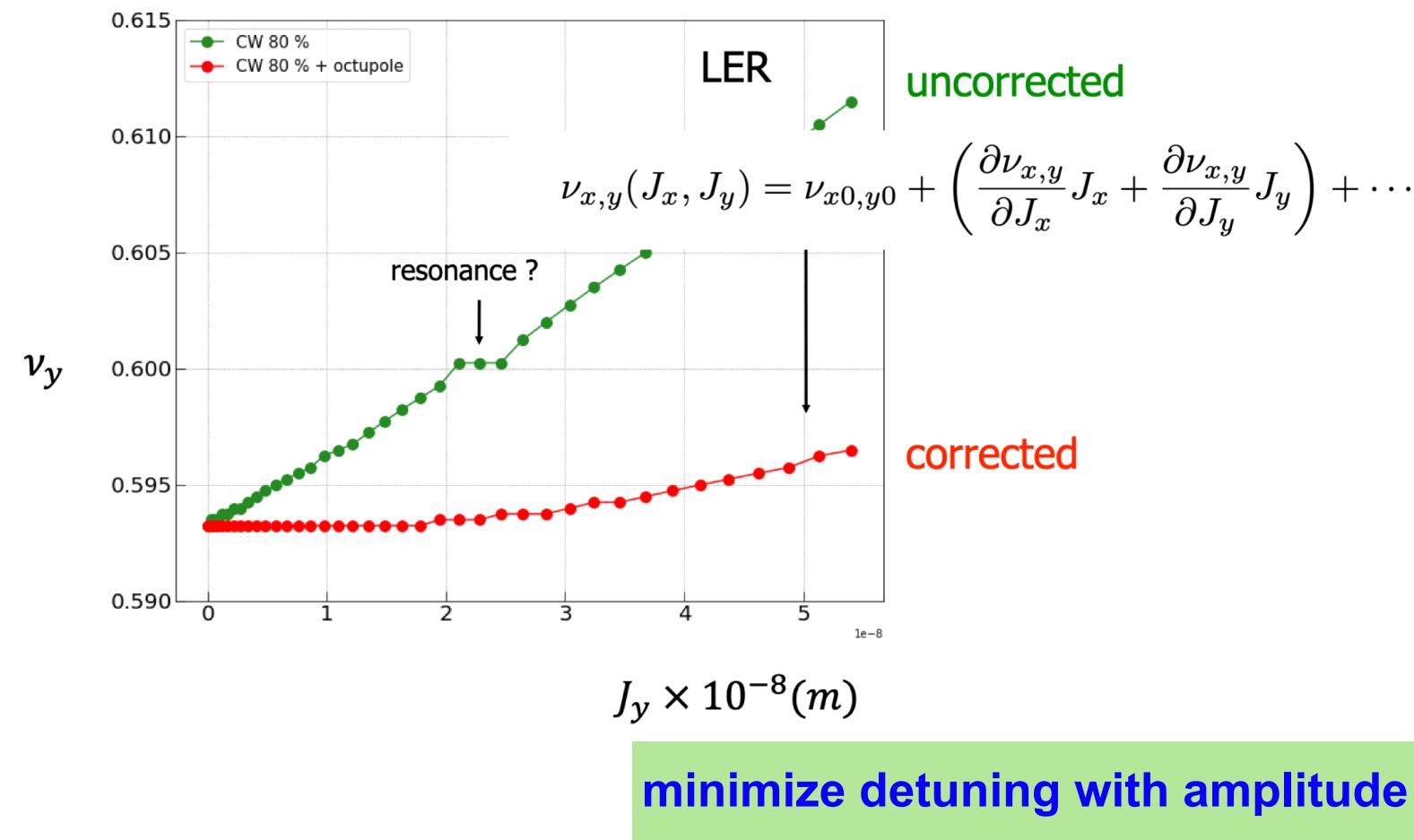
minimize detuning with amplitude

SOME THOUGHTS ON NON-LINEAR OPTIMIZATION

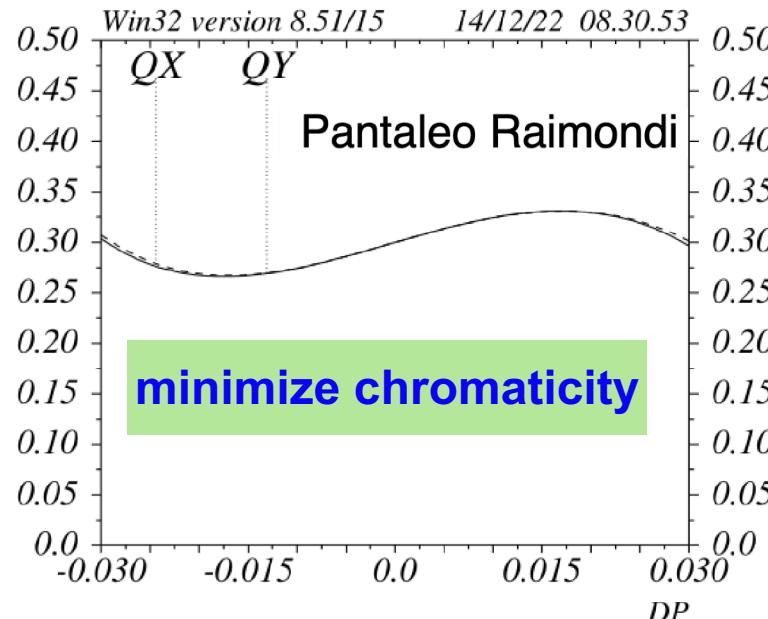


The ARC+LSS has zero second order chromaticity in both planes, only third and higher orders remain.

Y. Ohnishi / KEK

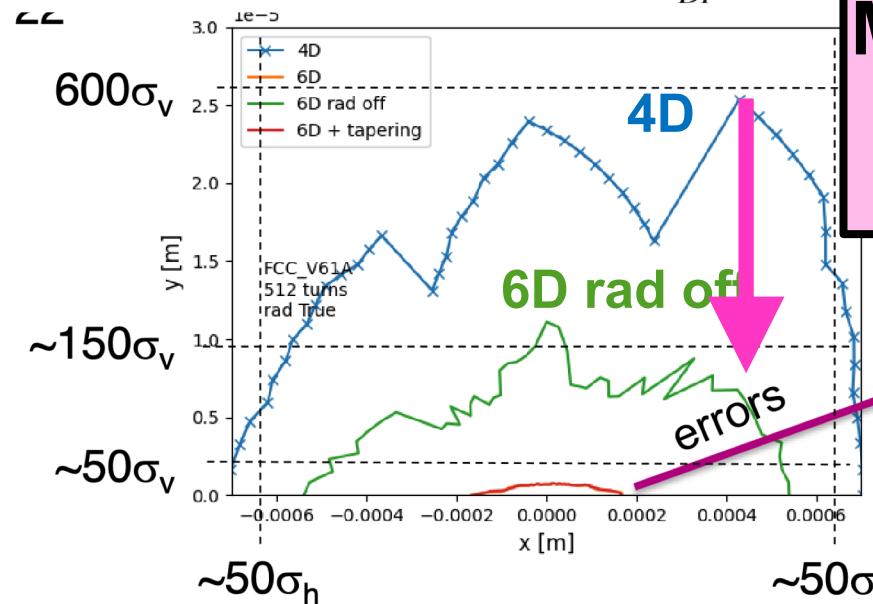
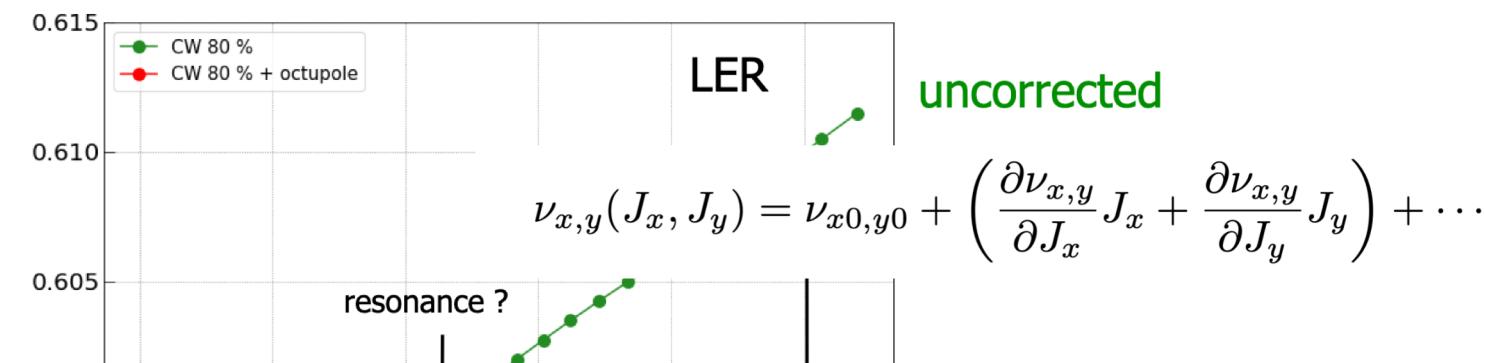


SOME THOUGHTS ON NON-LINEAR OPTIMIZATION



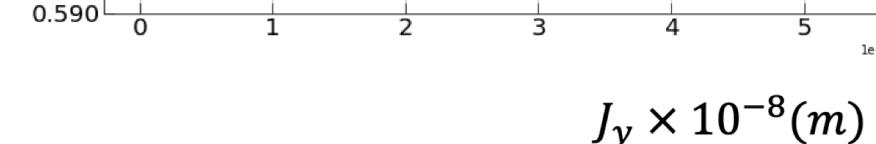
The ARC+LSS has zero second order chromaticity in both planes, only third and higher orders remain.

Y. Ohnishi / KEK

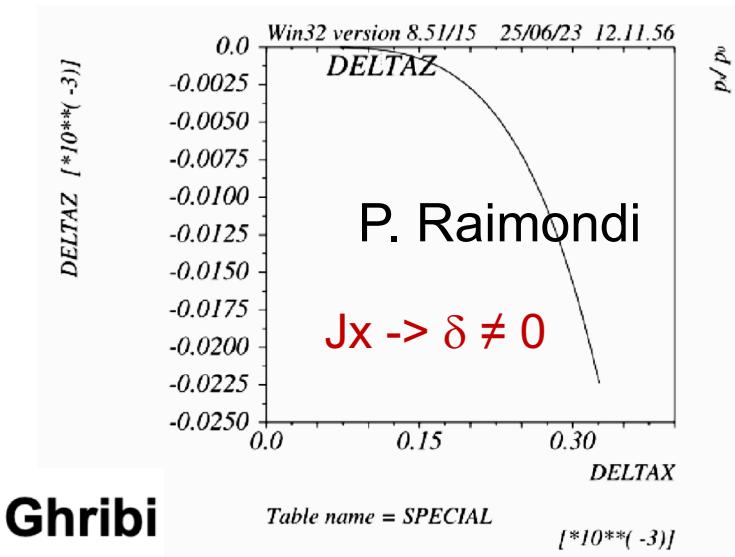
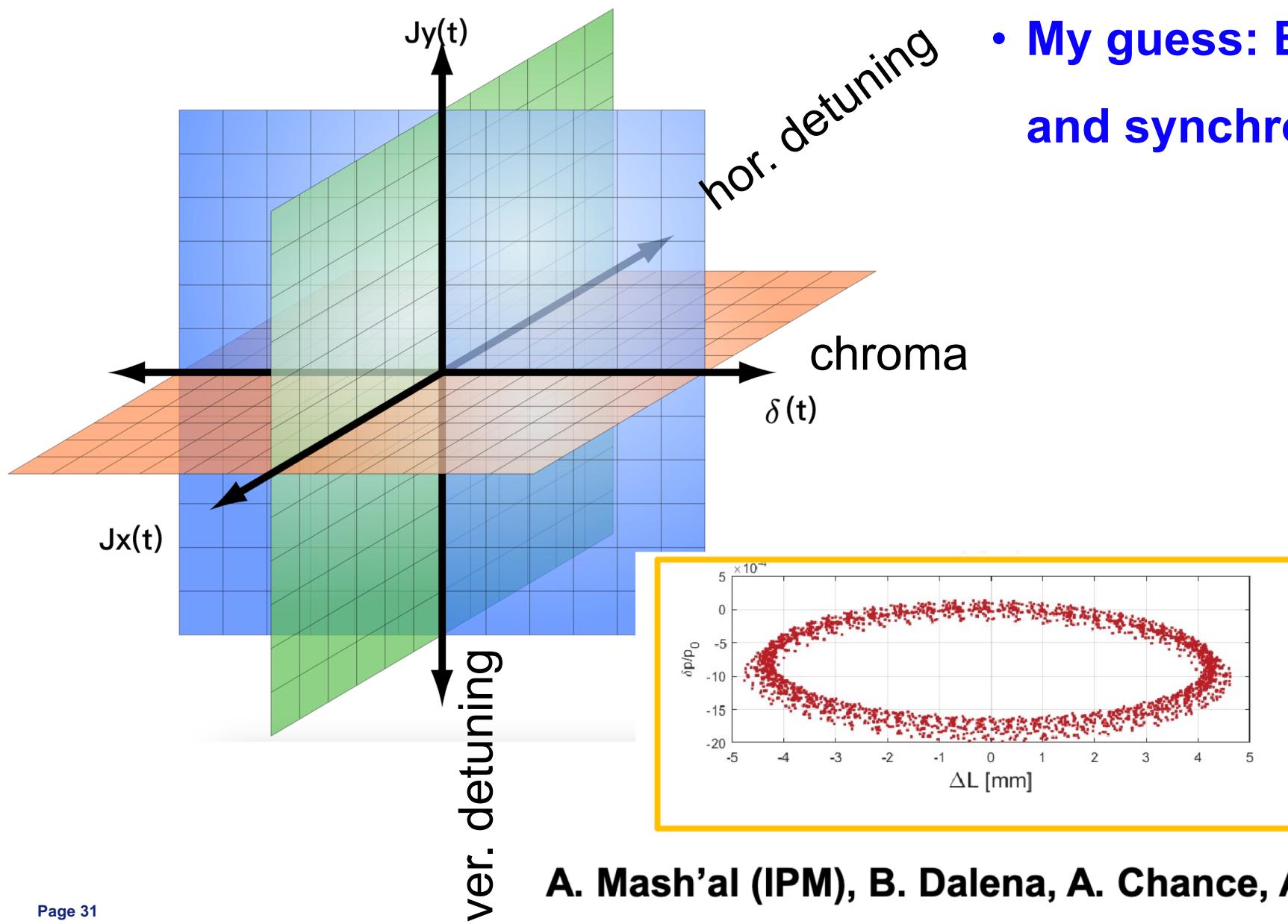


Simone Liuzzo

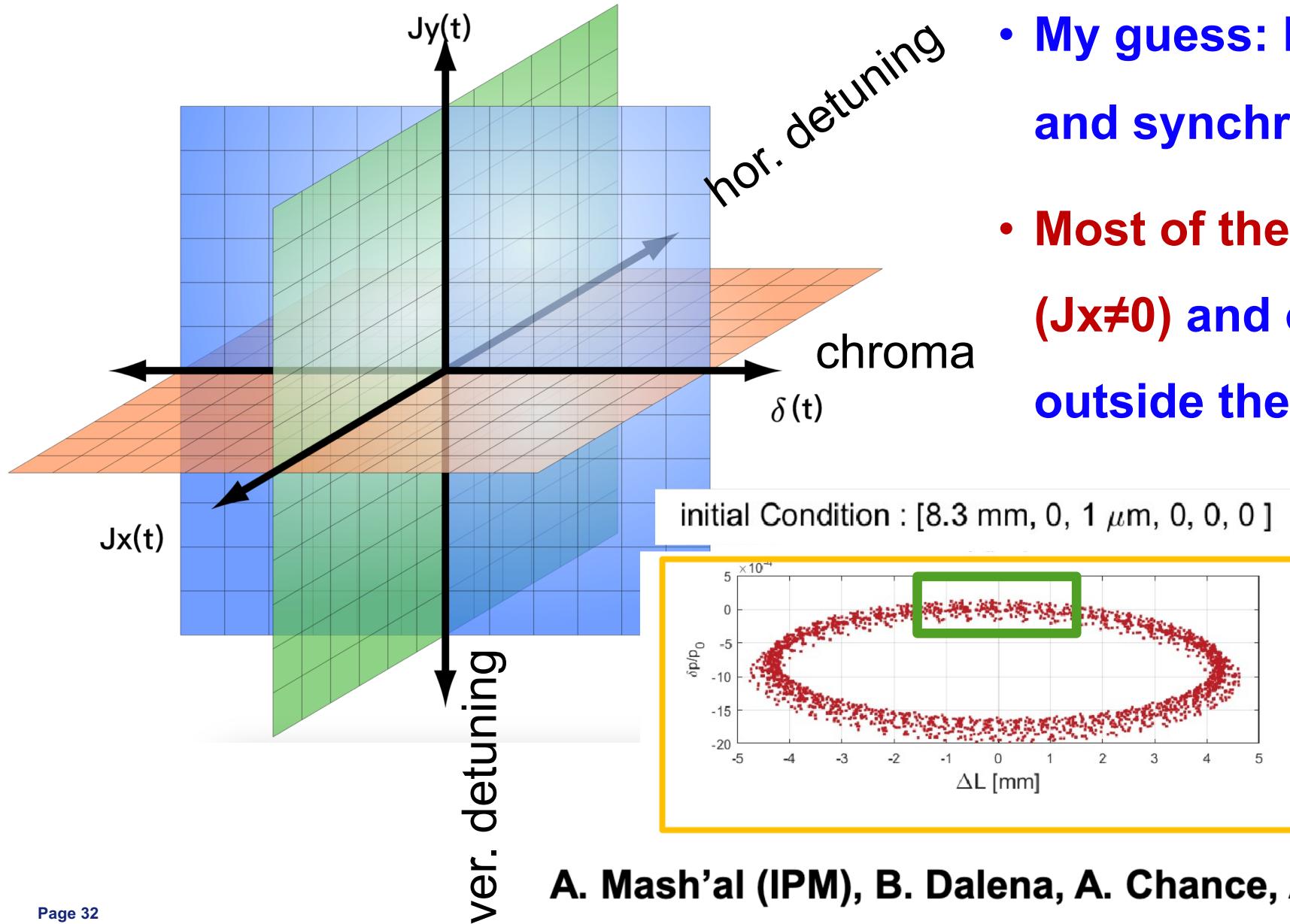
My old-standing puzzle: Where does all this reduction of DA come from?



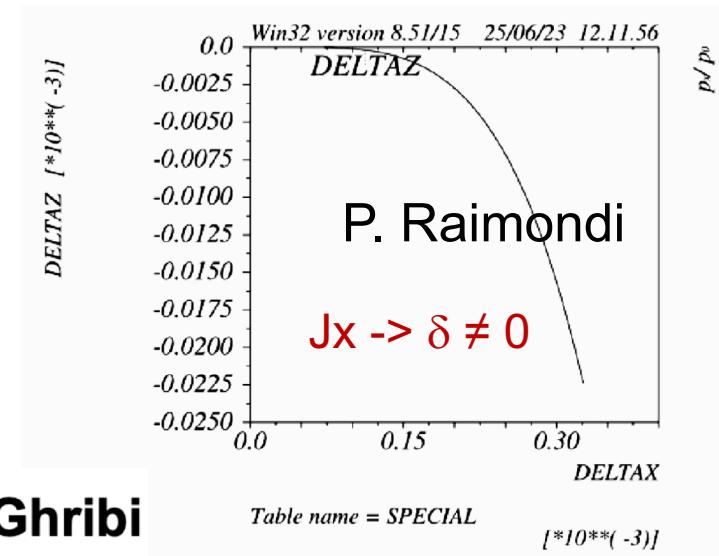
SOME THOUGHTS ON NON-LINEAR OPTIMIZATION



SOME THOUGHTS ON NON-LINEAR OPTIMIZATION



- My guess: Blame path lengthening and synchrotron motion for that.
- Most of the time they will be off-axis ($J_x \neq 0$) and off-energy ($\delta \neq 0$), thus, outside the 3 axis of the tune space.



SOME THOUGHTS ON NON-LINEAR OPTIMIZATION

Consequence: Minimizing cross-term (betatron+chromatic) detuning terms could be as much, if not more, effective than minimizing purely higher-order betatron or chromatic terms

ANHX, ANHY

$n(\varepsilon_1), n(\varepsilon_2),$
 $n(\delta_p)$

PTC_NORMAL,

$$Q(J_x(t), \delta(t)) = Q(J_x(0), \delta(0)) +$$
$$\left. \frac{\partial Q}{\partial J_x} \right|_{\delta=0} J_x(t) + \left. \frac{\partial Q}{\partial \delta} \right|_{J_x=0} \delta(t) +$$
$$\frac{1}{2} \left\{ \left. \frac{\partial^2 Q}{\partial J_x^2} \right|_{\delta=0} J_x^2(t) + \left. \frac{\partial^2 Q}{\partial \delta^2} \right|_{J_x=0} \delta^2(t) + \left. \frac{\partial^2 Q}{\partial J_x \partial \delta} \right| J_x(t) \delta(t) \right\} +$$

....

SOME THOUGHTS ON NON-LINEAR OPTIMIZATION

Blame path lengthening: Actually, minimizing linear chroma is still very helpful, since path lengthening and chroma are highly correlated

$$\Delta C = -2\pi(J_X \xi_X + J_Y \xi_Y). \quad (31)$$

$$Q(J_x(t), \delta(t)) = Q(J_x(0), \delta(0)) +$$
$$\frac{\partial Q}{\partial J_x} \Big|_{\delta=0} J_x(t) + \frac{\partial Q}{\partial \delta} \Big|_{J_x=0} \delta(t) +$$

PHYSICAL REVIEW SPECIAL TOPICS - ACCELERATORS AND BEAMS 8, 094001 (2005)

Dependence of average path length betatron motion in a storage ring

Yoshihiko Shoji*

SOME THOUGHTS ON NON-LINEAR OPTIMIZATION

Blame path lengthening: Actually, minimizing linear chroma is still very helpful, since path lengthening and chroma are highly correlated

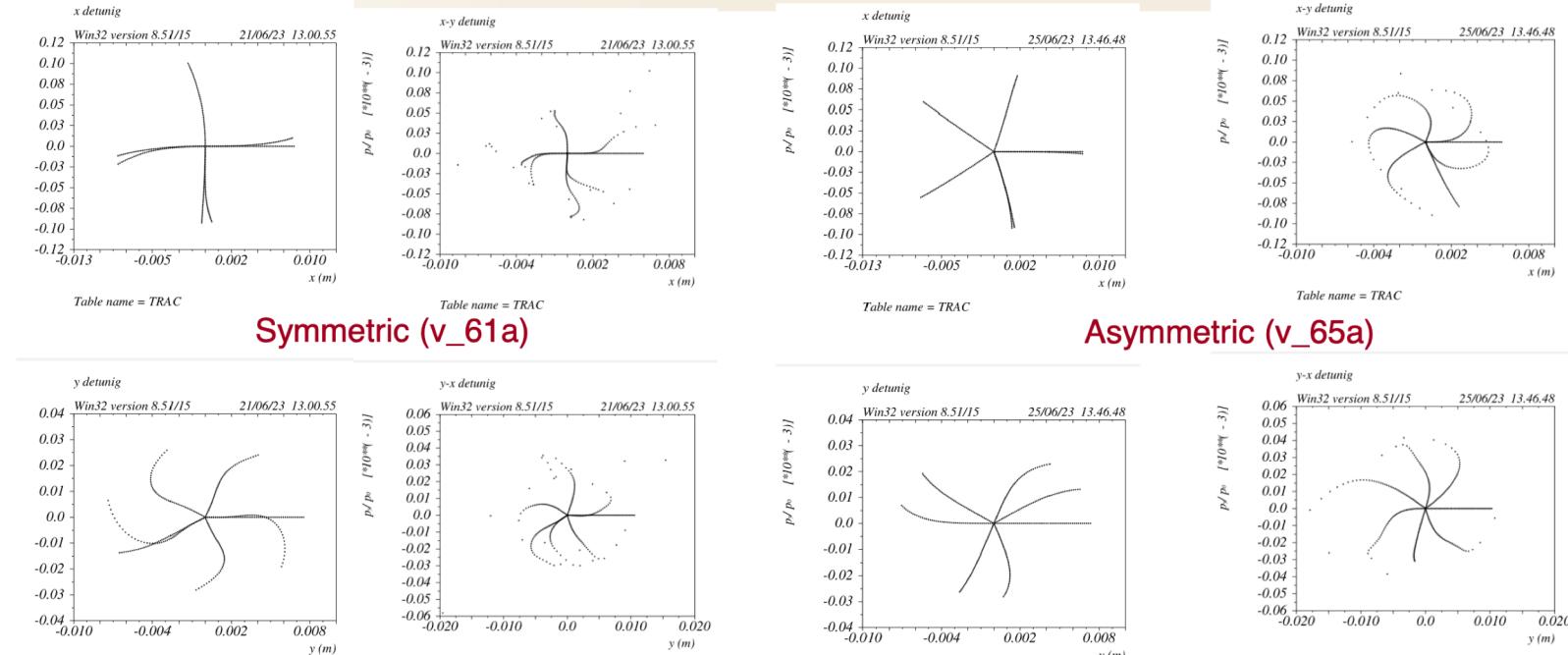
$$\Delta C = -2\pi(J_X \xi_X + J_Y \xi_Y) + O(\xi^2, J^2)$$

Maybe, higher-order terms might include higher-order chromatic terms.

$$\begin{aligned} Q(J_x(t), \delta(t)) &= Q(J_x(0), \delta(0)) + \\ &\quad \left. \frac{\partial Q}{\partial J_x} \right|_{\delta=0} J_x(t) + \left. \frac{\partial Q}{\partial \delta} \right|_{J_x=0} \delta(t) + \\ &\quad \frac{1}{2} \left\{ \left. \frac{\partial^2 Q}{\partial J_x^2} \right|_{\delta=0} J_x^2(t) + \left. \frac{\partial^2 Q}{\partial \delta^2} \right|_{J_x=0} \delta^2(t) + \left. \frac{\partial^2 Q}{\partial J_x \partial \delta} \right|_{J_x=0} J_x(t) \delta(t) \right\} + \end{aligned}$$

SOME THOUGHTS ON NON-LINEAR OPTIMIZATION

- betatron amplitude-dependent phase space deformation is induced by both detuning with amplitude and amplitude-dependent beta beating



Detuning from crab sextupoles further reduced.

$$\frac{\partial Q}{\partial J_x}, \frac{\partial \beta}{\partial J_x}, \frac{\partial \alpha}{\partial J_x}, \frac{\partial \gamma}{\partial J_x}$$

minimize detuning with amplitude