Ion Simulations as a Guide to FT-ICR Performance
Progress in supercomputer modeling of ion dynamics in Penning and Kingdon traps and quadrupols

Eugene N. Nikolaev,

Institute for Energy Problems in Chemical Physics
Institute for Biochemical Physics,
Russian Academy of Sciences, Moscow

Challenges of FT ICR MS

•Further improvement of the FT ICR mass accuracy and resolution

•Increasing dynamic range of FT ICR mass spectrometry

Possible Solutions:

Higher magnetic field

Exploration of other than Penning trap designs

What is really going on inside FT ICR cell?!

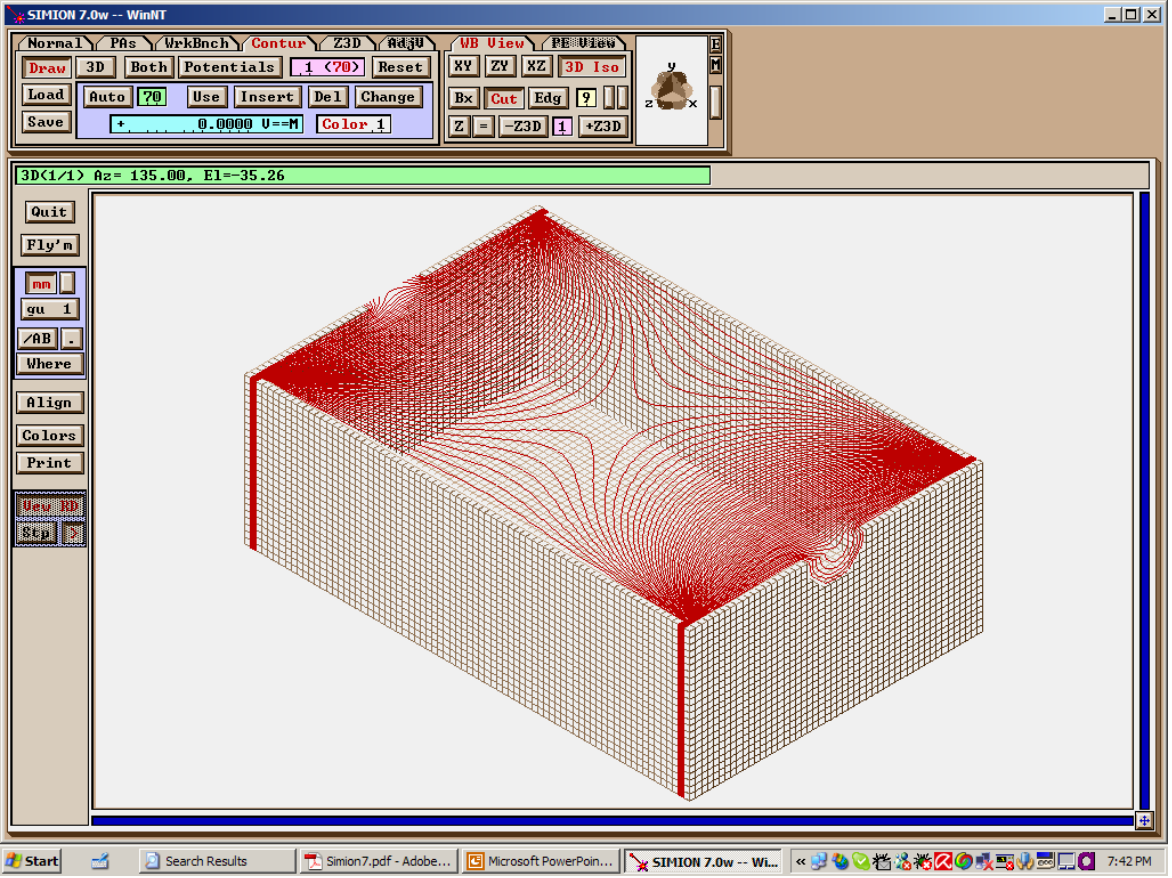
How can we investigate ion motion there?

As we know from experiment, ion-ion interaction is the key point

From the view point of physics ion clouds in FT ICR cell form charged plasma

The main approach is a computer simulation of ion motion dynamics

What kind of computational tools do we have?



INEEL-95/0403
Formerly EGG-CS-7233
Rev. 5

SIMION 3D VERSION 7.0 USER'S MANUAL

David A. Dahl

Published 2000

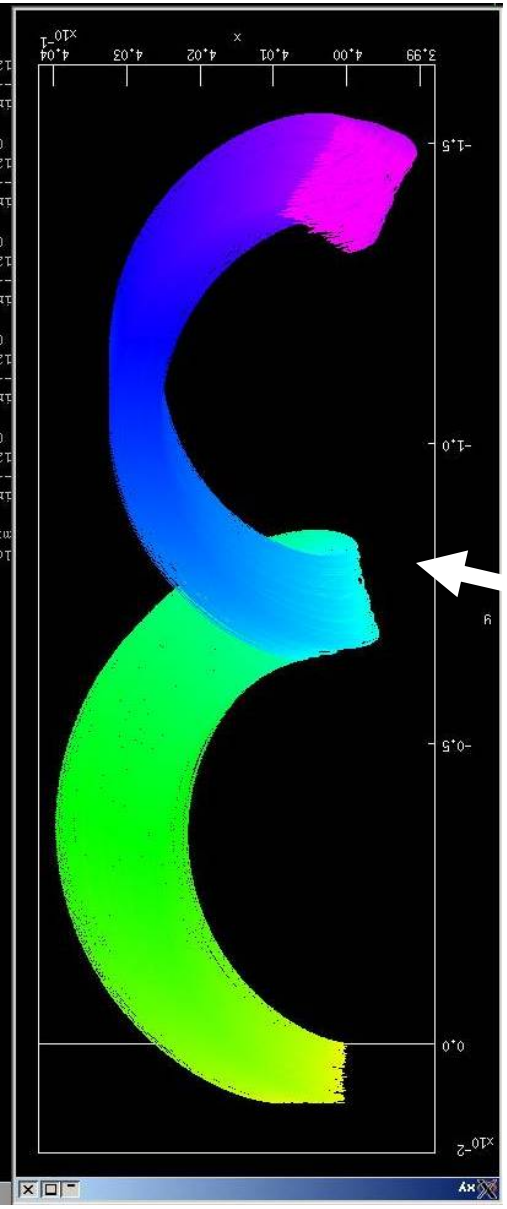
Idaho National Engineering and Environmental Laboratory
Idaho Falls, ID 83415

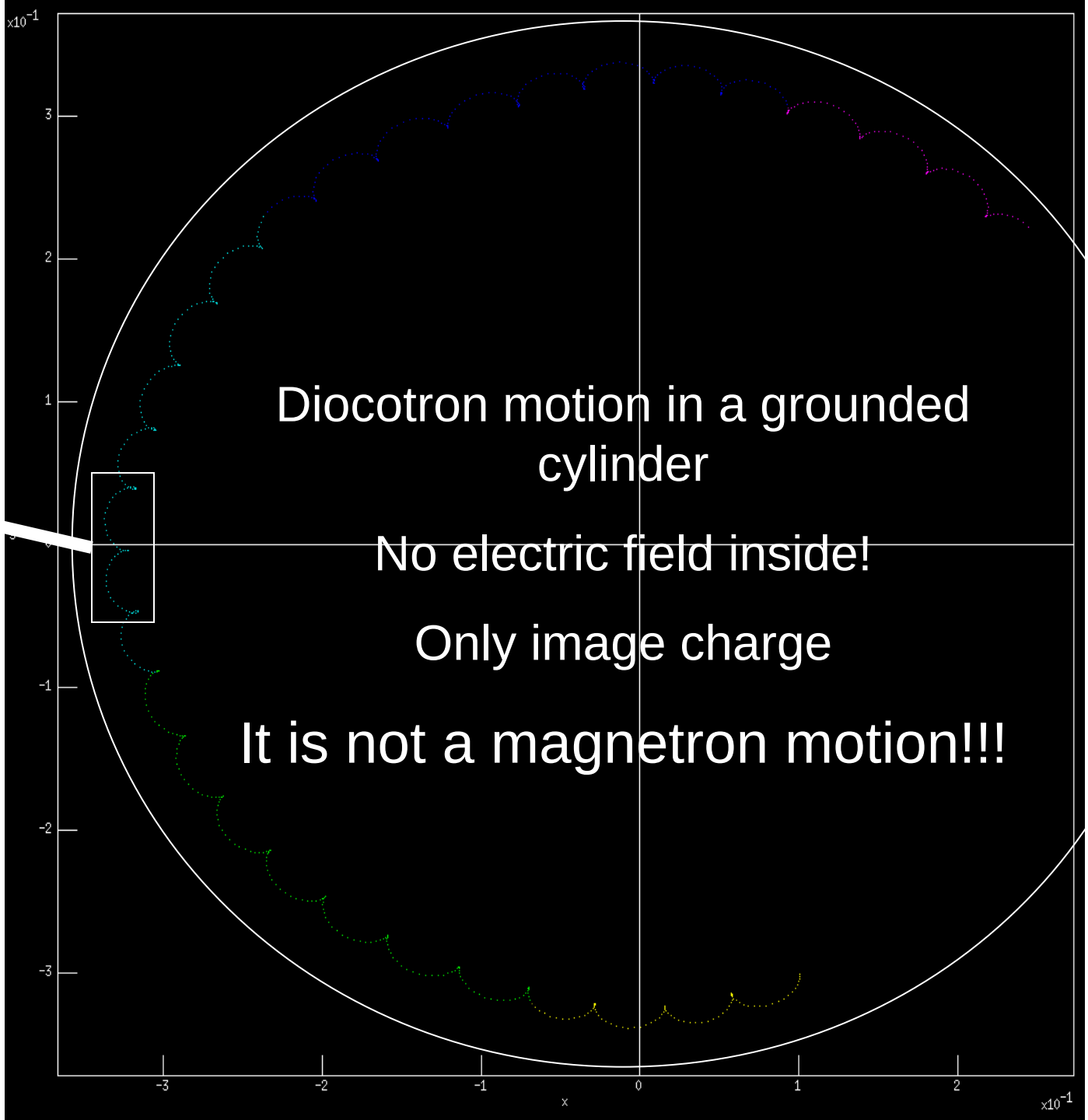
Prepared for the
U. S. Department of Energy
Office of Energy Research
Under DOE Idaho Operations Office
Contract DE-AC07-99ID13727

What SIMION can not do:

- Can not take properly ion-ion interaction and so can not explain ion behavior in accumulation devices and ICR cell for instance

- Can not take into account ion-image charge interaction





Diocotron motion in a grounded cylinder

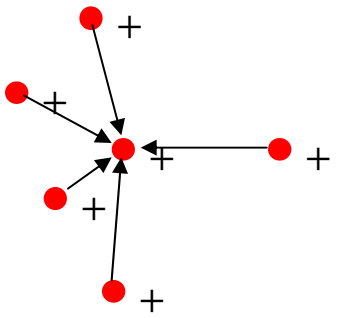
No electric field inside!

Only image charge

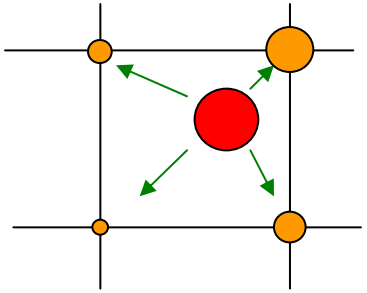
It is not a magnetron motion!!!

Two main methods of multi particle dynamic simulation in “charged plasma”

- The Particle-Particle Method



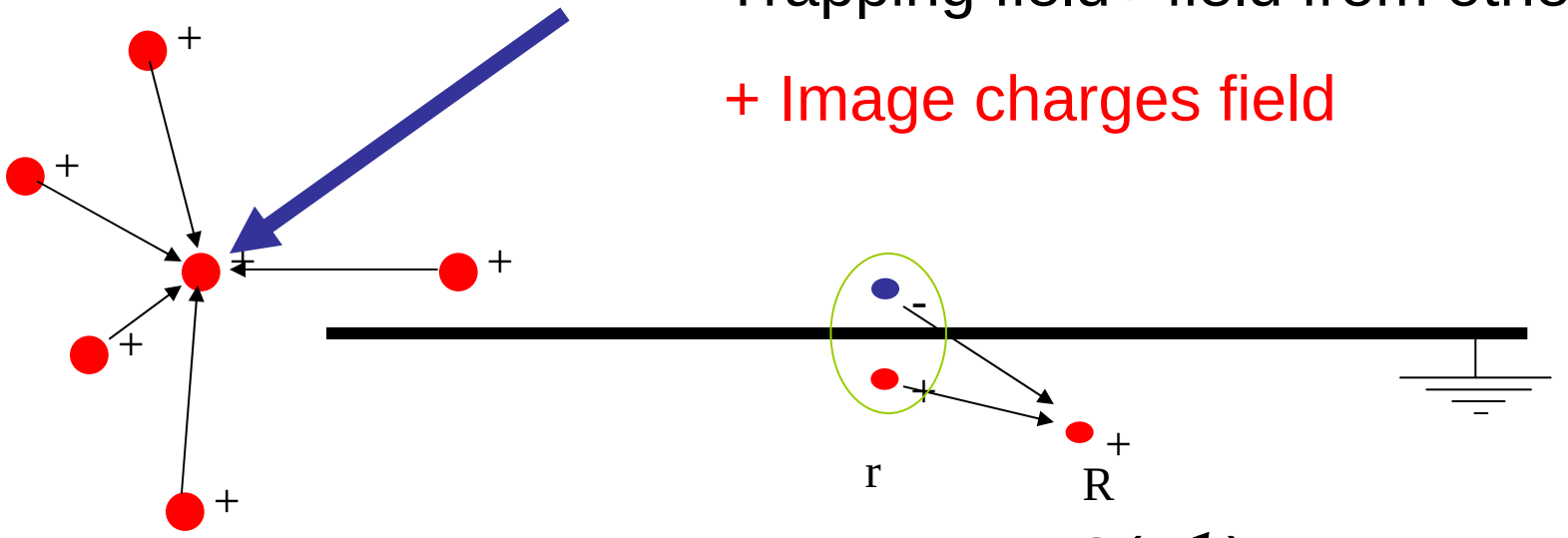
- The Particle-Cell(Mesh) Method



Forces acting on an ion inside ICR trap

Trapping field+ field from other ions

+ Image charges field



$E=G(r,\vec{R})$ Green function

Lorenz force

Green function is very complicated

Even for simple geometry ICR cells!!

The Particle-Particle Method

Miluchihin N.V.; Miura, K., Inoue, M.

Application of a Parallel Computer to Simulation of Ion Trajectories in an Ion Cyclotron Resonance Spectrometer

Rapid Commun. Mass. Spectrom. 7(1993) 966-970.

E.N.Nikolaev, N.V.Miljuchihin, Masao Inoe;

Evolution of ion cloud in a Fourier transform ion cyclotron resonance mass spectrometer during signal detection: its influence on spectral line shape and position

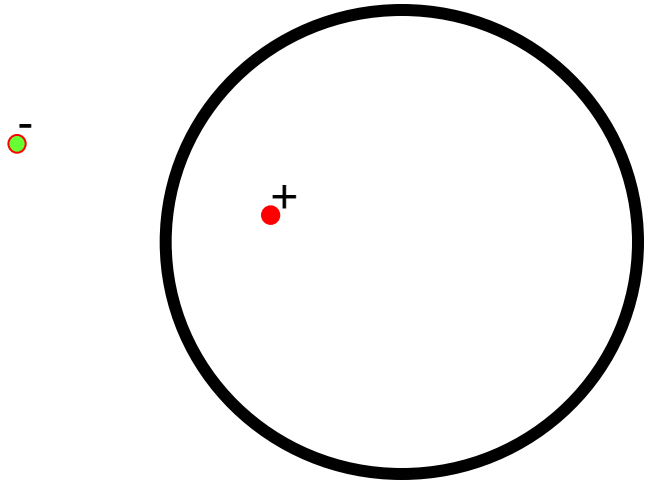
Int. J. Mass Spec. Ion Proc. 148 (1995) 145-157

The problem of particle-particle method:

How to account for ion-image charge interaction?

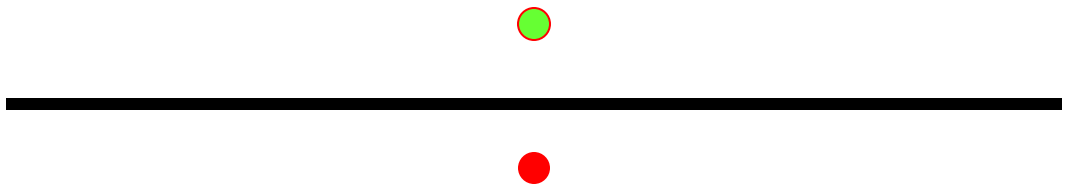
Simple analytical expression for Green function is known for

Sphere



and

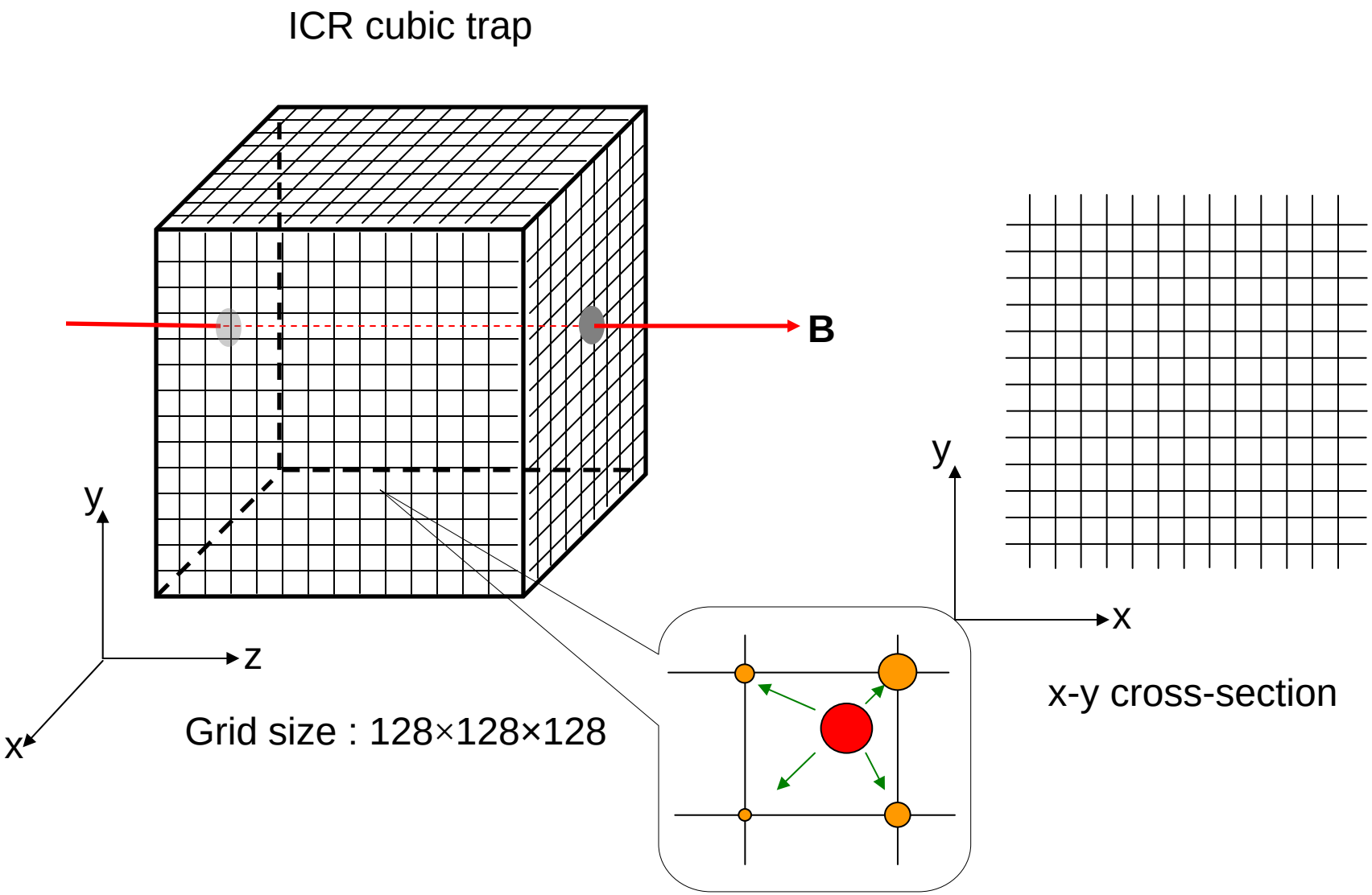
Infinite surface



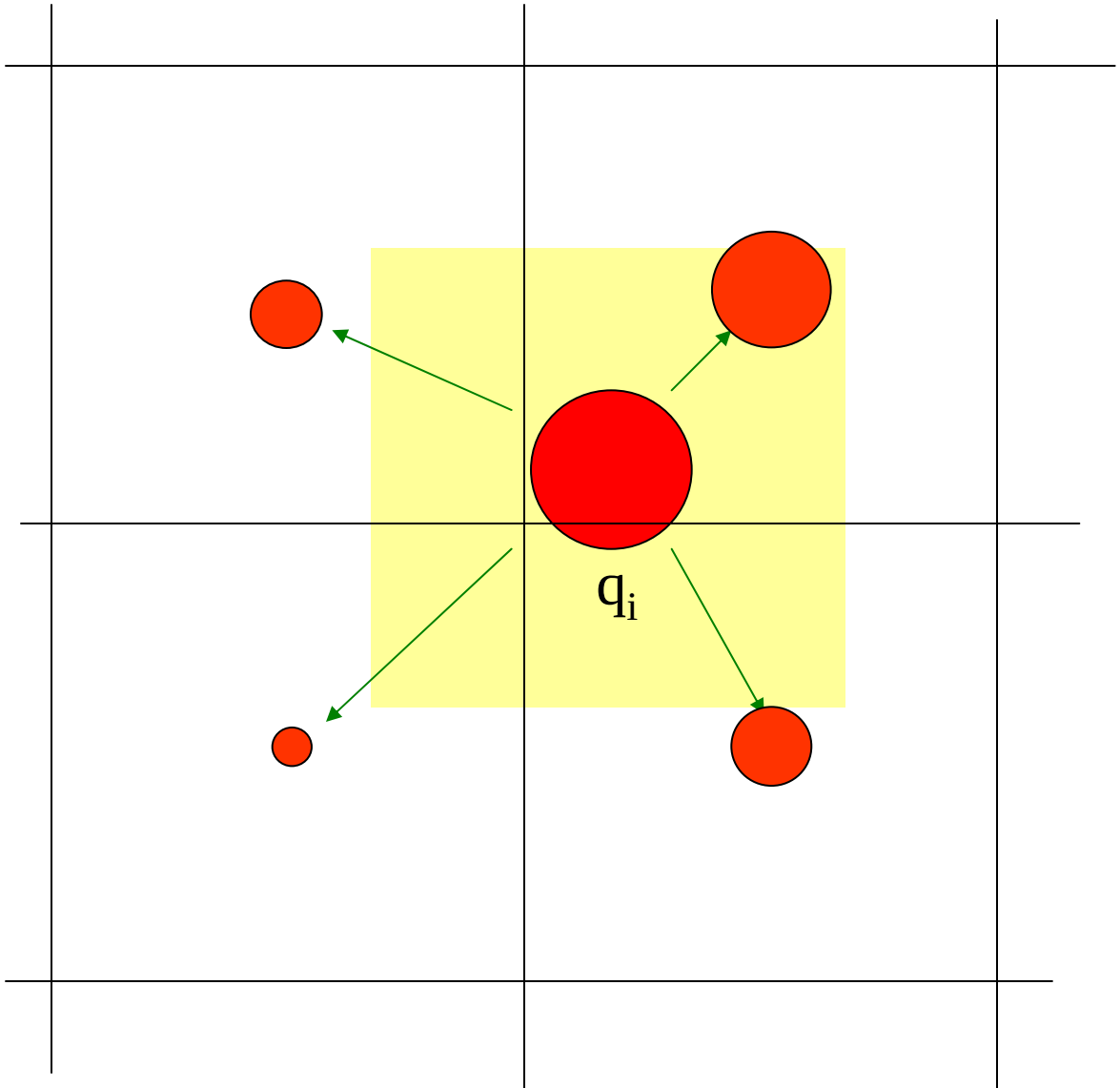
Particle in Cell or Particle in Mesh algorithm

Dale Mitchell, Dick Smith

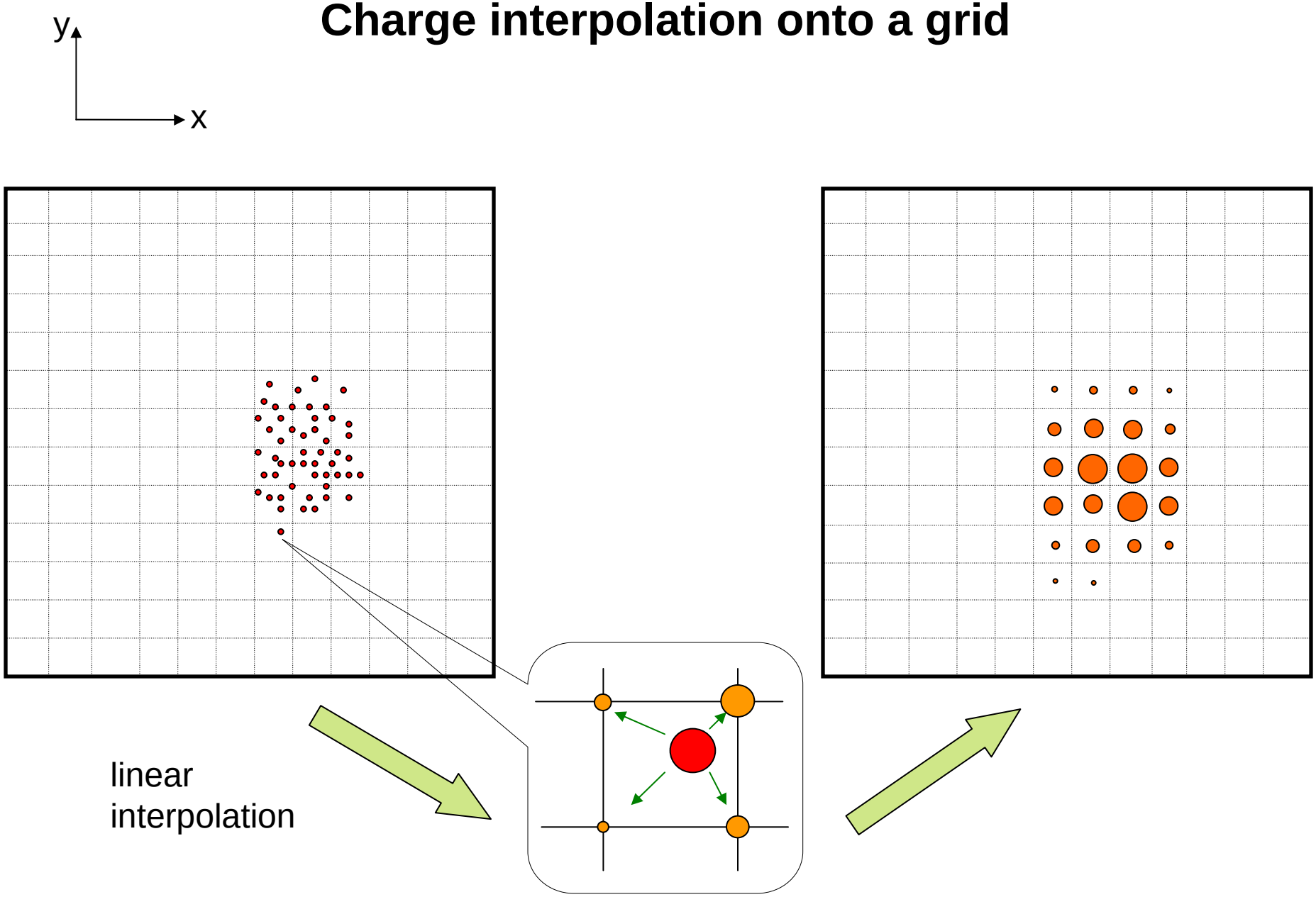
Particle-In-Cell Algorithm (first used to describe ion behavior in FT ICR cell by Dale Mitchell)



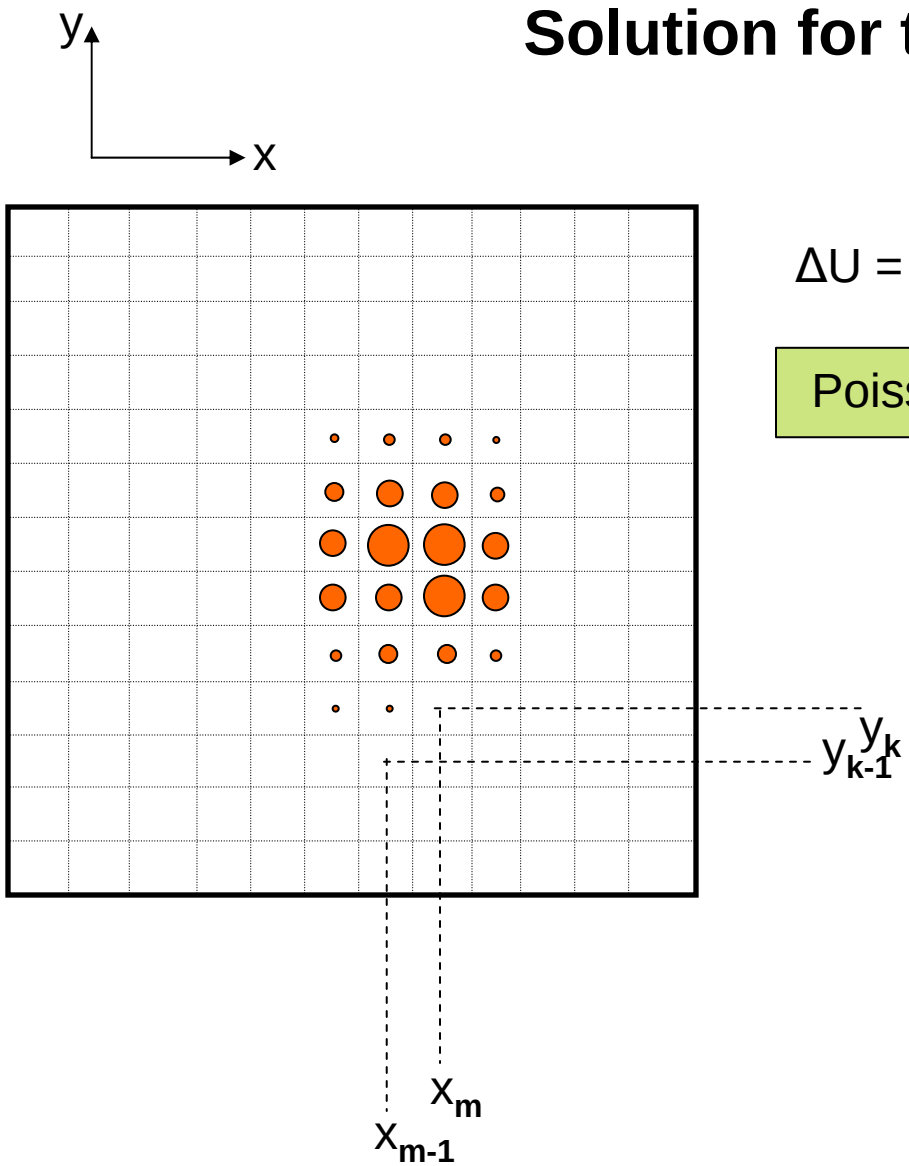
“Weighted” charge assignment to the nearest grid points



Charge interpolation onto a grid



Solution for the electric field

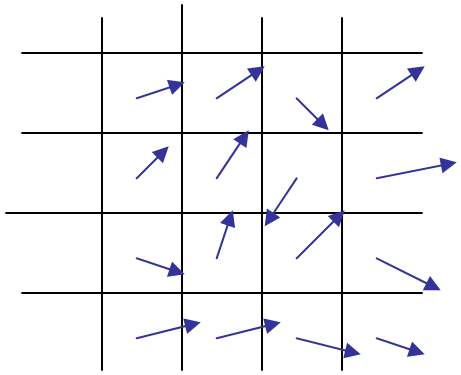


$$\Delta U = -\rho(x_i, y_j, z_k)/\varepsilon_0$$

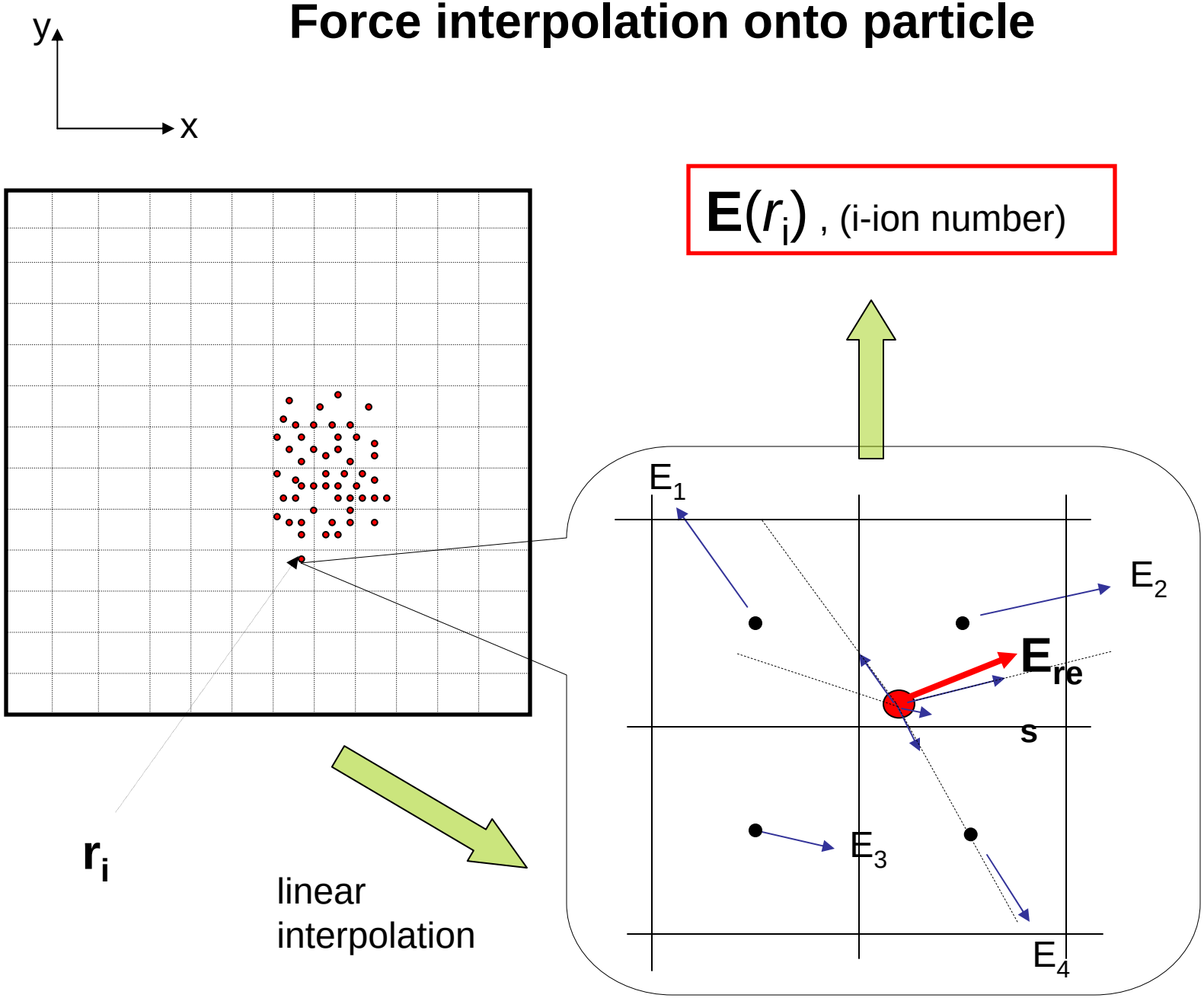
Poisson Solver

$$\mathbf{U}(x_i, y_j, z_k) \quad i, j, k=1\text{-}p$$

$$\boxed{\mathbf{E}(x_i, y_j, z_k) \quad i, j, k=1\text{-}p}$$



Force interpolation onto particle



Integration of the equation of motion

$$m\frac{d^2\vec{r}}{dt^2} = q\vec{E} + q\left(\frac{d\vec{r}}{dt}\times\vec{B}\right)$$

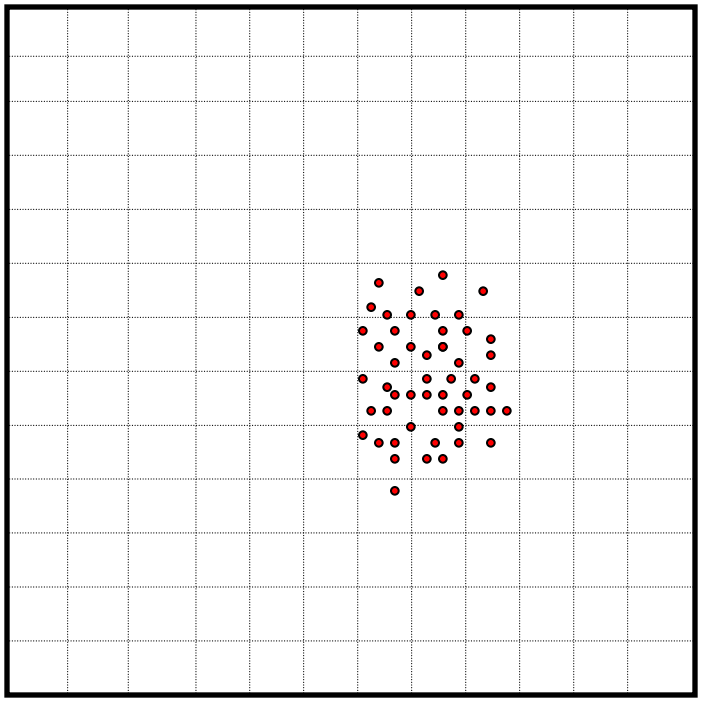
$\mathbf{E}(r_i)$

v_i , r_i

Boris integrator

for every ion (i)

new v_i , r_i



Next time step

Integration of the equation of motion

$$m\frac{d^2\vec{r}}{dt^2} = q\vec{E} + q\left(\frac{d\vec{r}}{dt}\times\vec{B}\right)$$

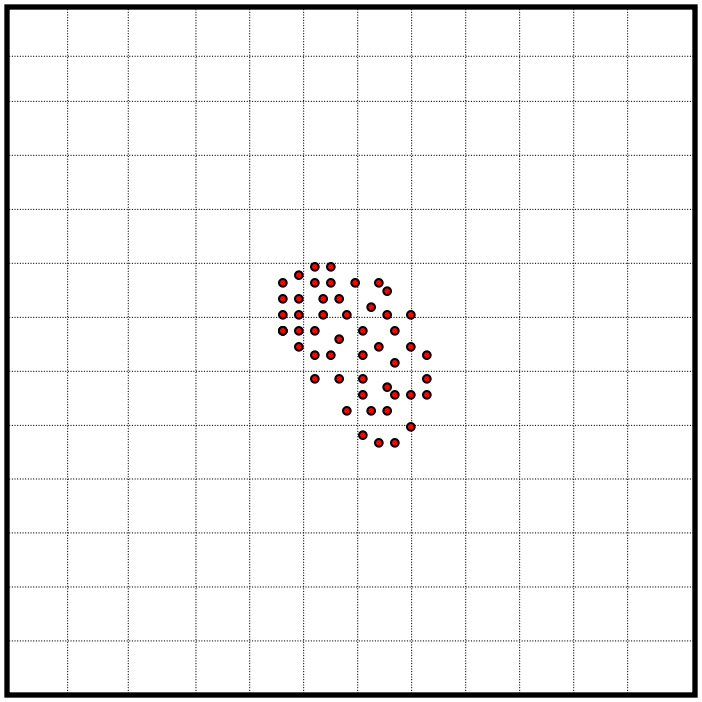
$\mathbf{E}(r_i)$

v_i , r_i

Boris integrator

for every ion (i)

new v_i , r_i



Next time step

Computational time

The Particle-Particle Method (without taking into account image charge interaction)		The Particle-Mesh Method
$10N_p^2 - N_p$	Number of operations per **one time-step** for coulombically interacting particles	$\alpha N_p + \beta(N)$
~ 1 day	CPU time for **one time-step** for N=32, $N_p=10^5$	**~ 50 ms**

N_p – number of particles

N – number of mesh points

$\alpha \sim 10$

$\beta \sim N^3 \log_2 N^3$ for an (N×N×N) mesh

Computers

Machine	*CPU*	*Memory*	*Architecture*	*Compiler*	*OS*
Regatta (p690)	16 Power4 (1.1 GHz)	64 GB shared memory	SMP (symmetric multiprocessor) system	XL Fortran for AIX	IBM AIX 5L v5.2
Solo (6 x p690)	4 p690 x 32 Power4 (1.1 GHz), 2 p690 x 32 Power4+ (1.7 GHz)	1 GB of memory per processor	Distributed SMP (symmetric multiprocessor) system	XL Fortran for AIX	IBM AIX

SGI Altix 3700 Aster system

Aster is an SGI Altix 3700 system, consisting of 416 CPUs (Intel Itanium 2, 1.3 GHz, 3 Mbyte cache each), 832 Gbyte of memory and 2.8 Tbyte of scratch disk space. The total peak performance is 2.2 Teraflop/sec.

The 416 processors are divided over 5 nodes: 4 batch nodes and 1 interactive node:

type	processors	Gbytes memory	Gbytes scratch	nodes
interactive	32	64	400	1
batch	64	128	400	2
batch	128	256	800	2

For 1650 ions in 4.7 T
623032 time steps
(around 10 ms real time of ion ensemble motion)
takes 3.5 hours on 4 processors

What was found?

In the case of conventional (not ideally hyperbolic) Penning trap there should be critical minimum number of ions in the trap to get a signal

Evolution of Z-compressed cloud for different ion densities

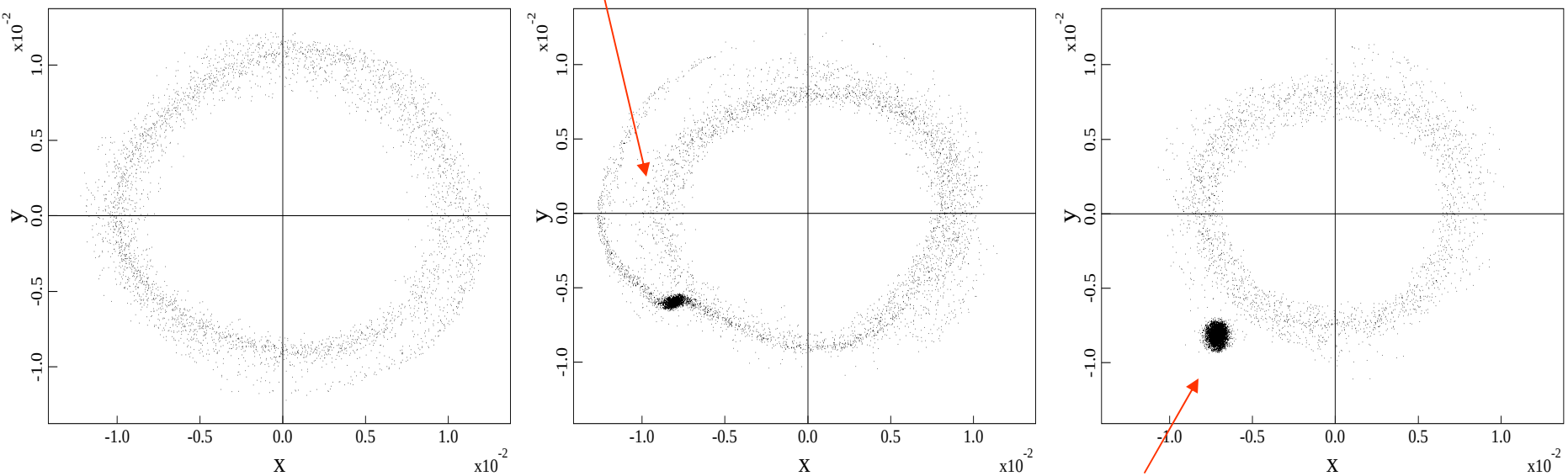
cloud initial radius 0.15 cm

cloud initial length 0.10 cm

T_{exc}= 0.07 ms , T_{detect}= 30 ms

Plasma instability!!

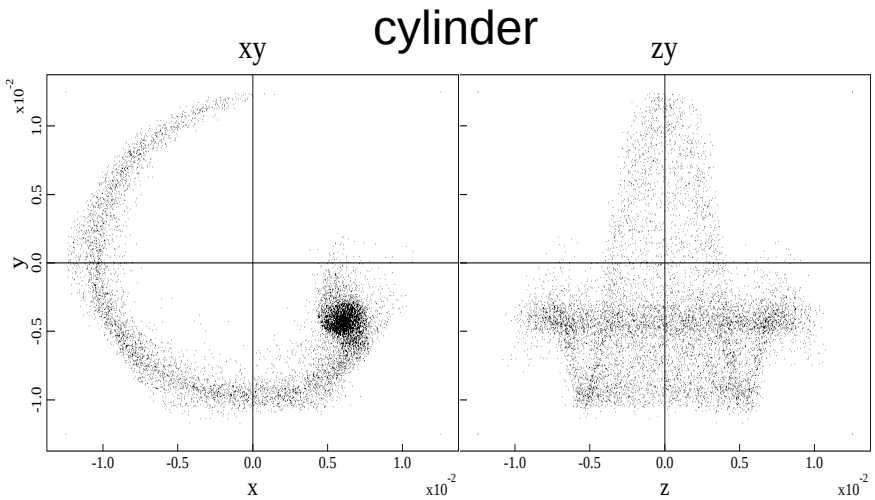
N=5000 **N=20000** **N=50000**

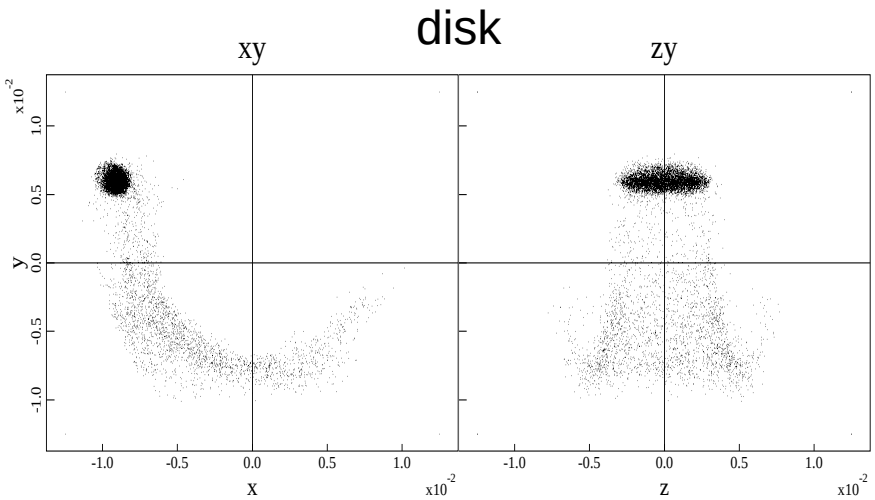


Phase locking

t = 8.10 ms

Ion cloud evolution (50000 ions)



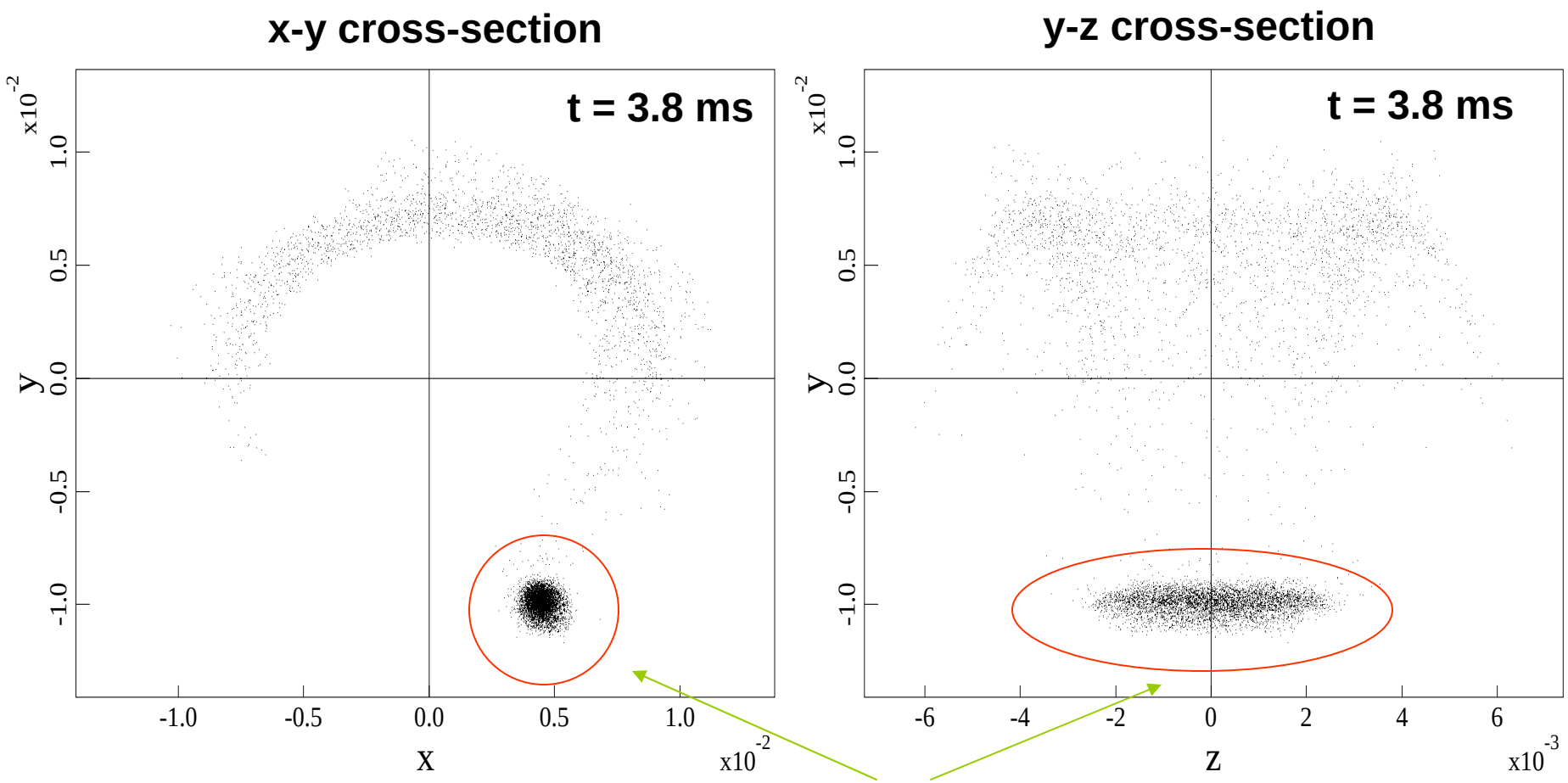


t = 2.34 ms

Ion clouds form comet like structures with condensation of a portion of ion clouds in the tale or the head of the comet

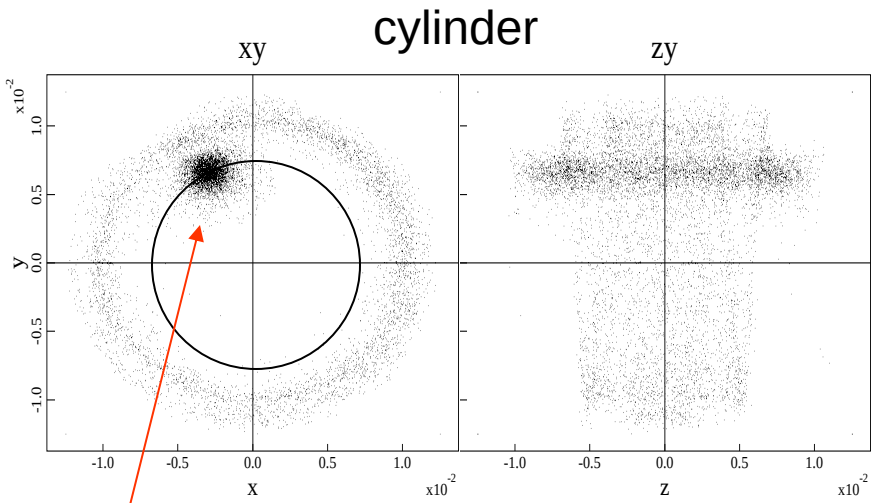
Separation of ion cloud into two ensembles: giving signal and giving noise (around 50:50)

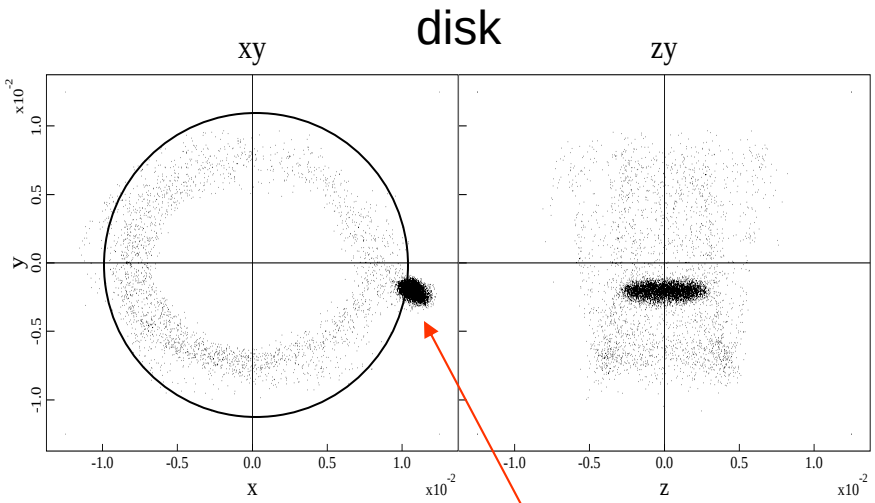
T_{exc} = 70 μs



Pronounced ion condensation or phase locking

Ion cloud evolution (50000 ions)





inside

outside

t = 6.24 ms

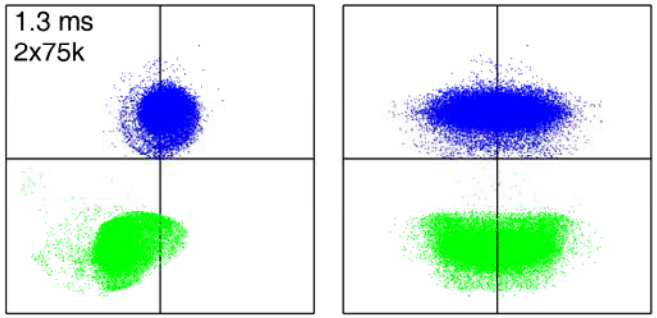
Condensed and spread parts of clouds have different cyclotron radii depending on the initial ion cloud forms

Coalescence (observed before)

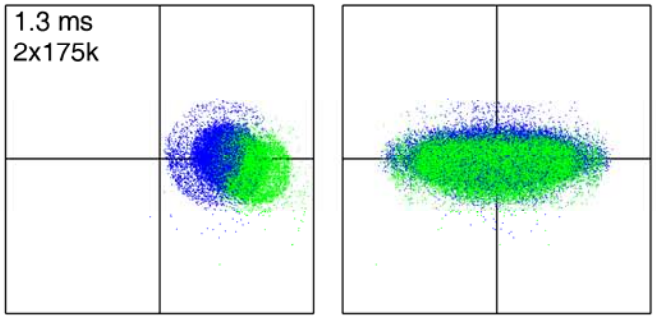
The behavior of two interacted ion clouds in ICR cell

m/z = 100.0 and 100.3
(equal number of ions)

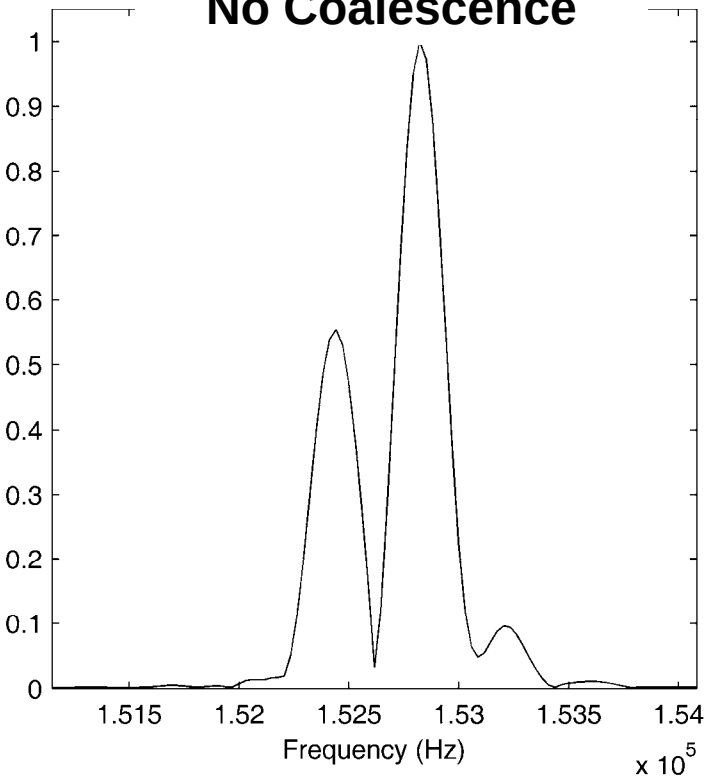
150,000 ions



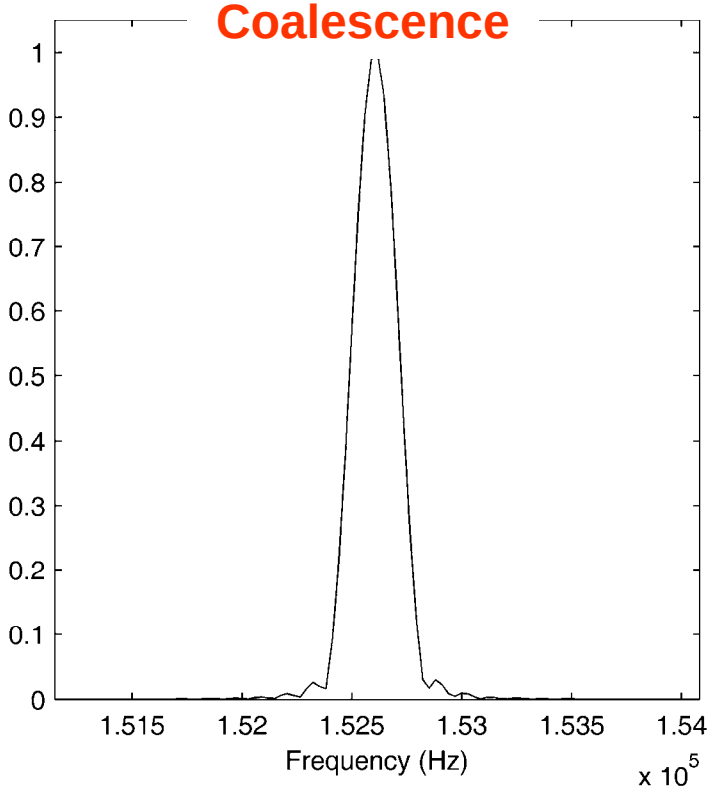
300,000 ions



No Coalescence



Coalescence



Behavior of multiple ion clouds

The behavior of three interacted ion clouds in ICR cell

m/z = **99.7**, **100.0**, **100.3**

initial form - z-compressed ellipsoid

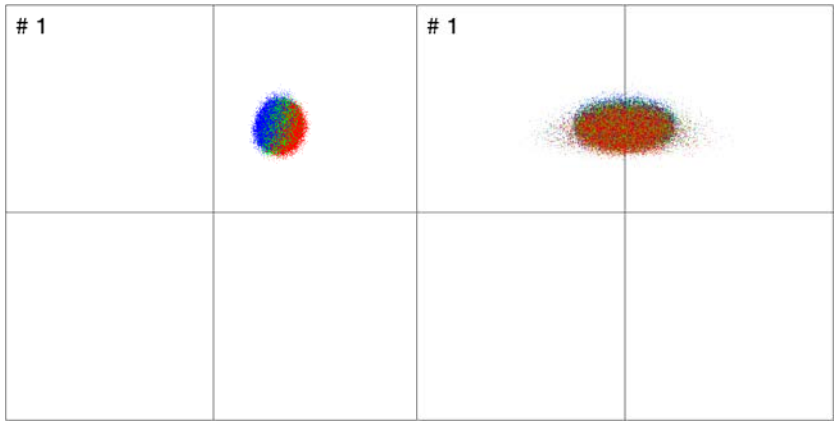
Excitation t = **45 μs**

peak-to-peak amplitude **= 10 V**

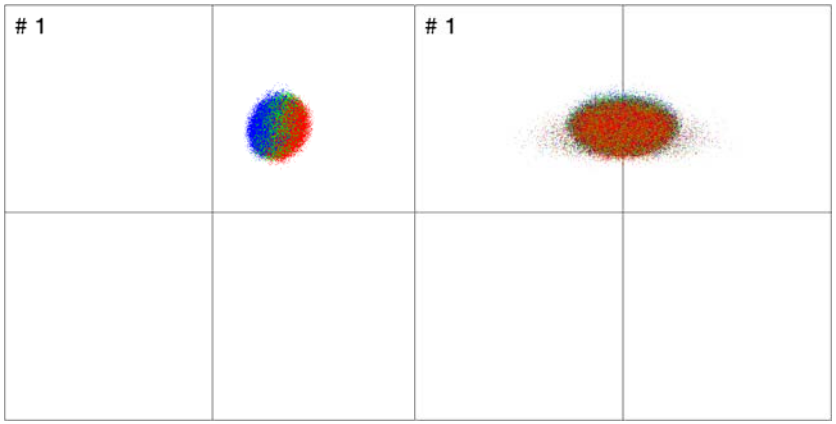
resonant to m/z **= 100.0**

After excitation period

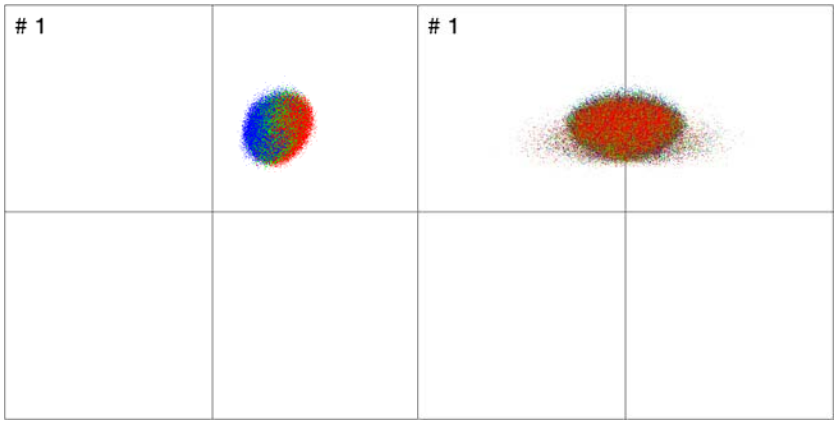
75,000 ions



150,000 ions



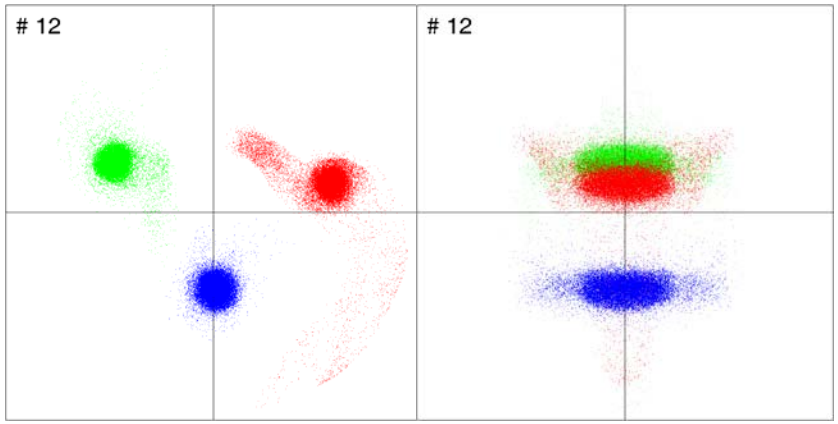
225,000 ions



- m/z = 99.7
- m/z = 100.0
- m/z = 100.3

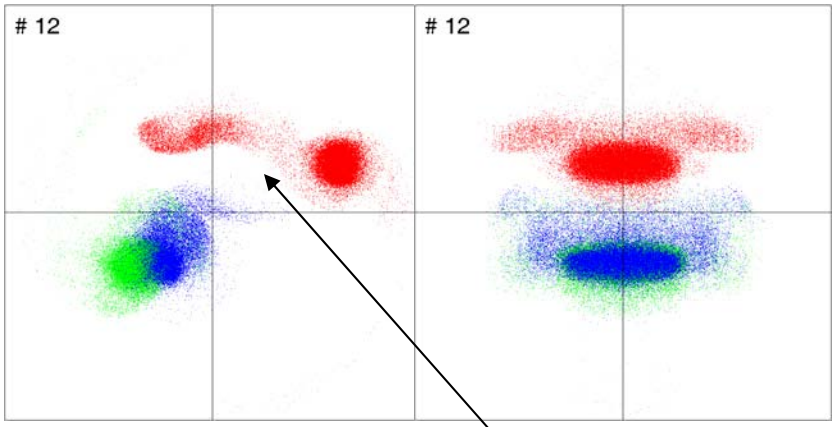
t = 02.93 ms

75,000 ions



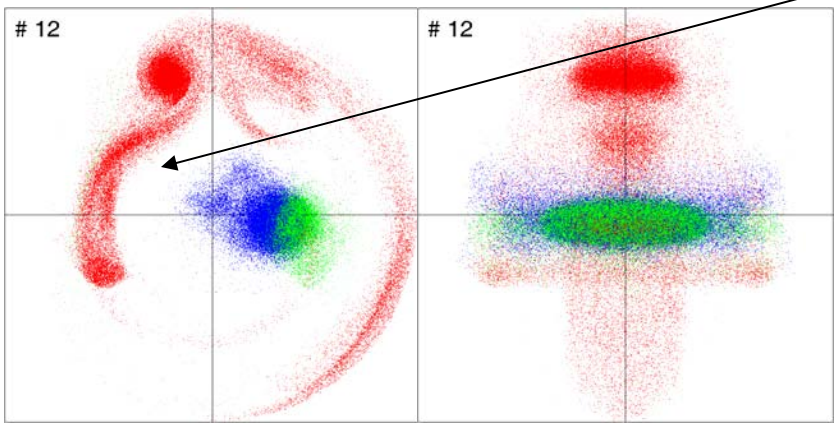
Independent motion

150,000 ions



Splitting into two clouds

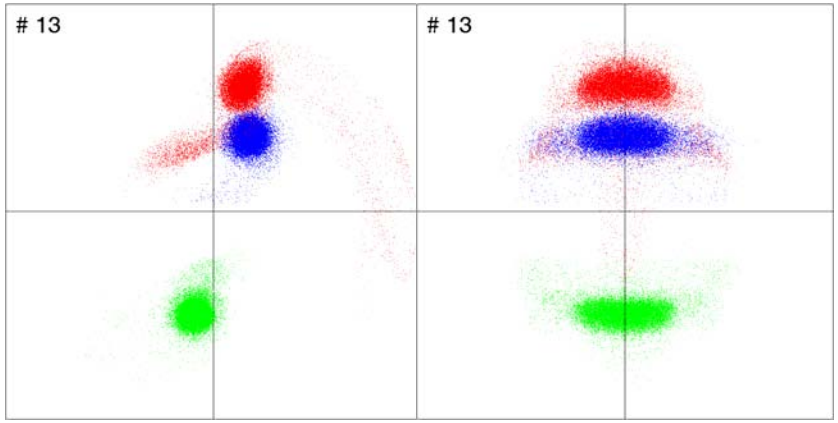
225,000 ions



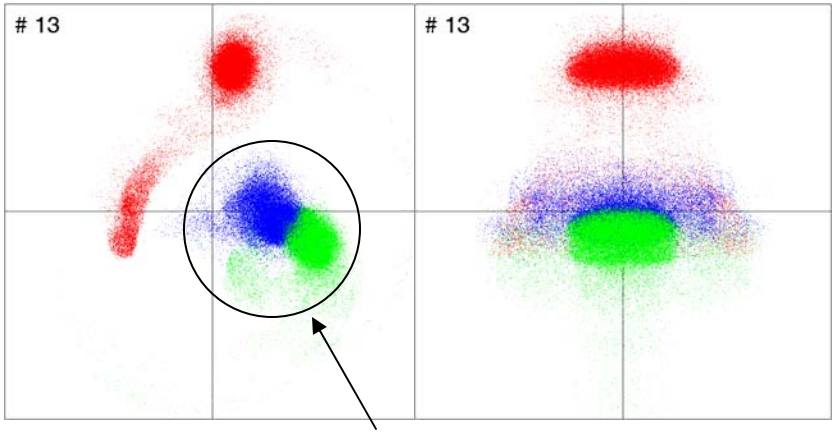
- m/z = 99.7
- m/z = 100.0
- m/z = 100.3

t = 03.19 ms

75,000 ions



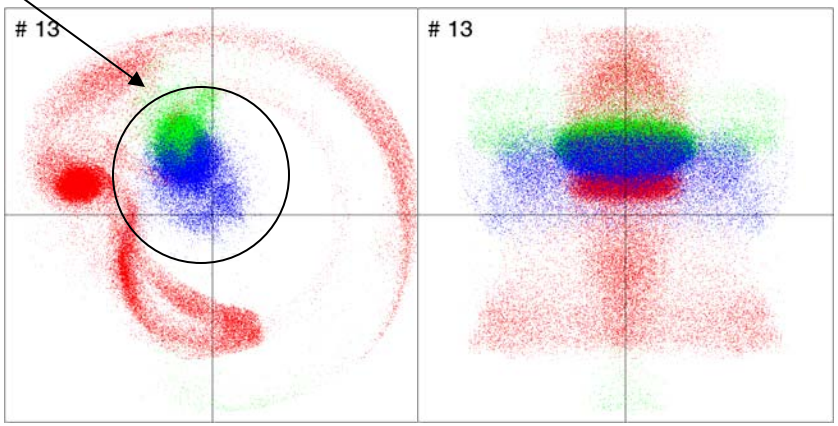
150,000 ions



Correlation

Complete frequency locking

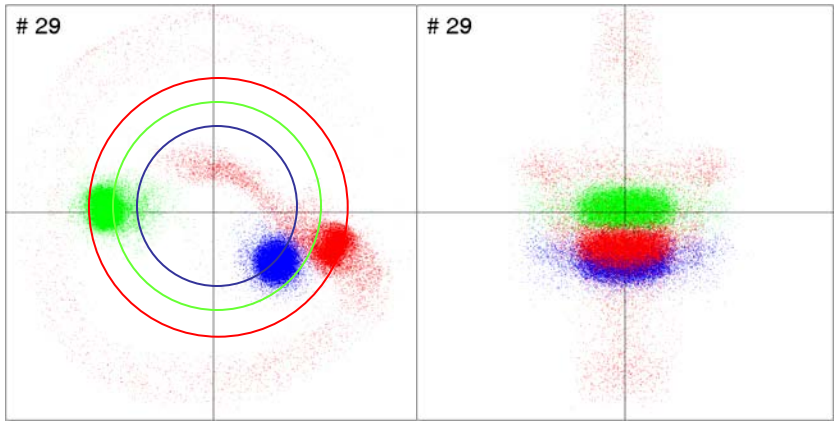
225,000 ions



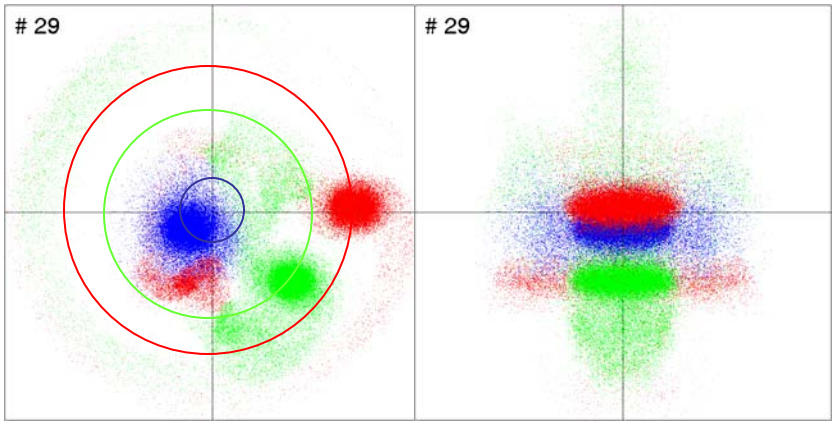
- m/z = 99.7
- m/z = 100.0
- m/z = 100.3

t = 07.45 ms

75,000 ions

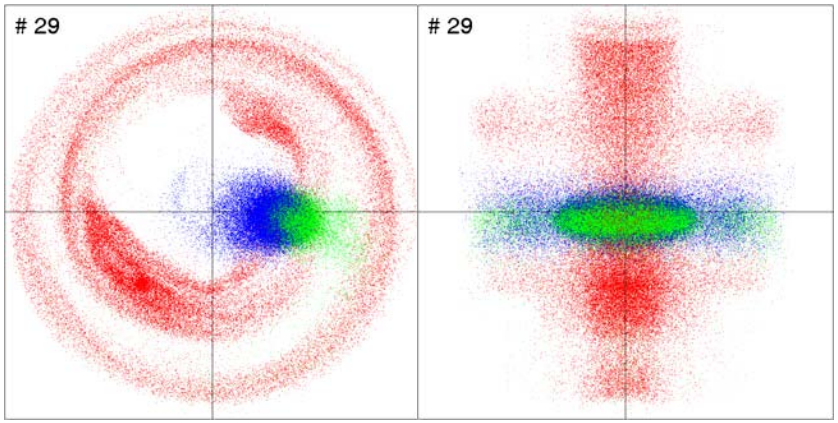


150,000 ions



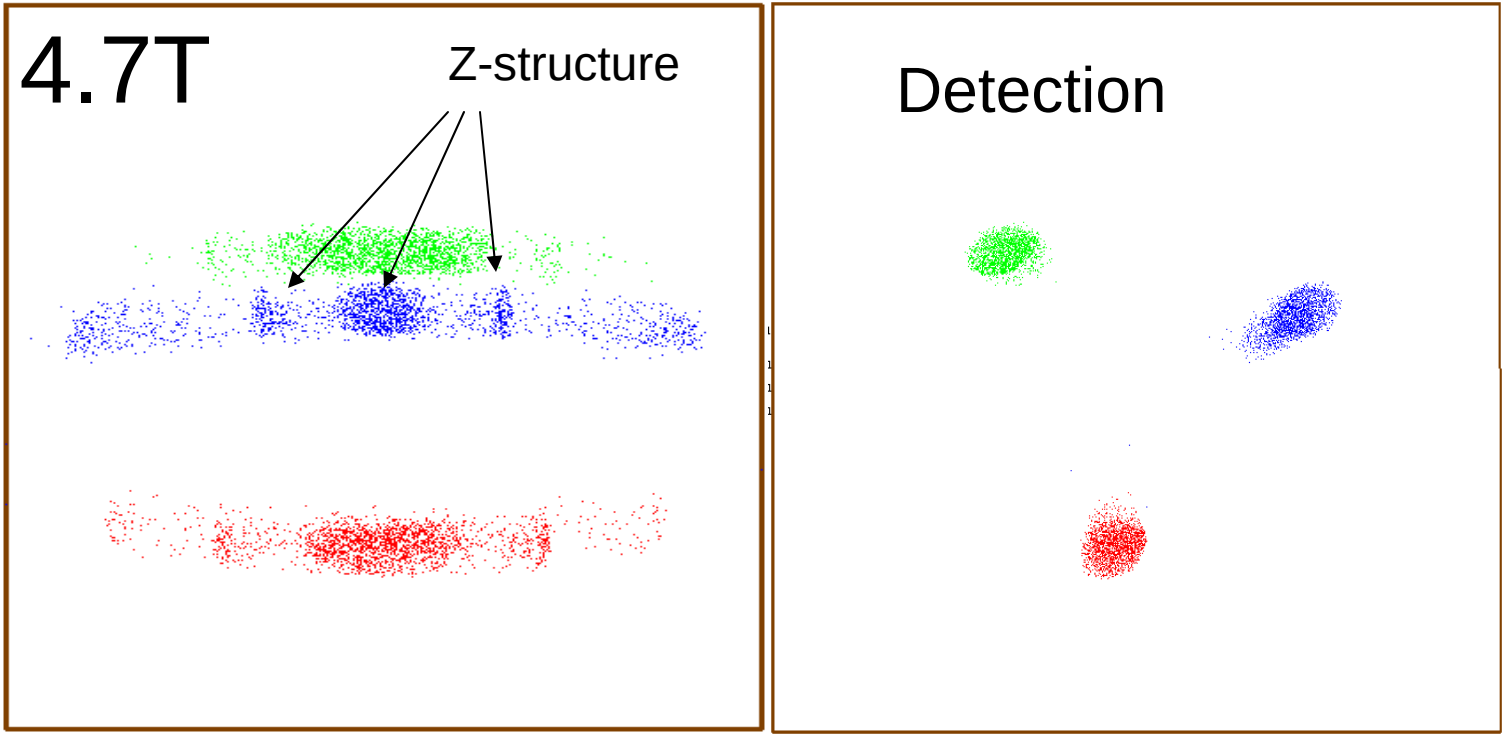
Different cyclotron radiuses !!!

225,000 ions

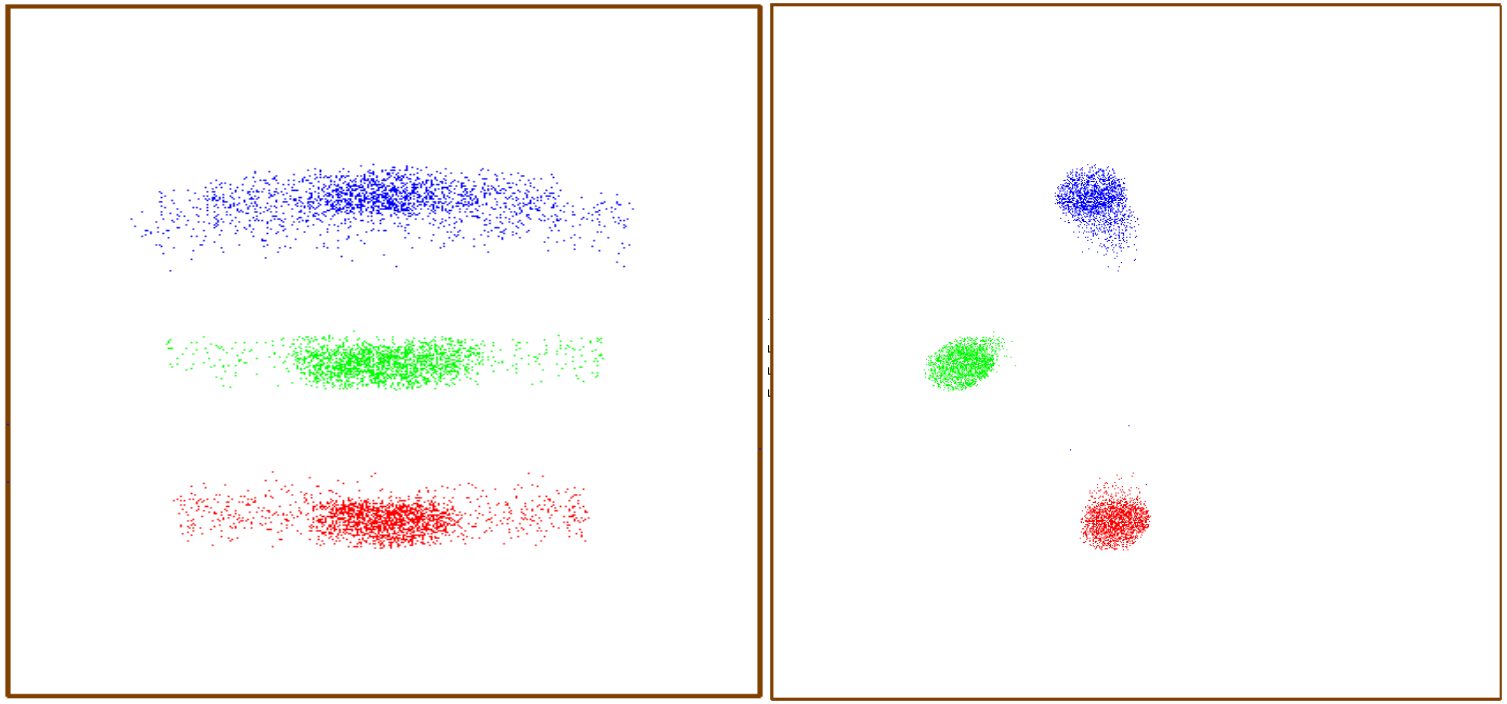


- m/z = 99.7
- m/z = 100.0
- m/z = 100.3

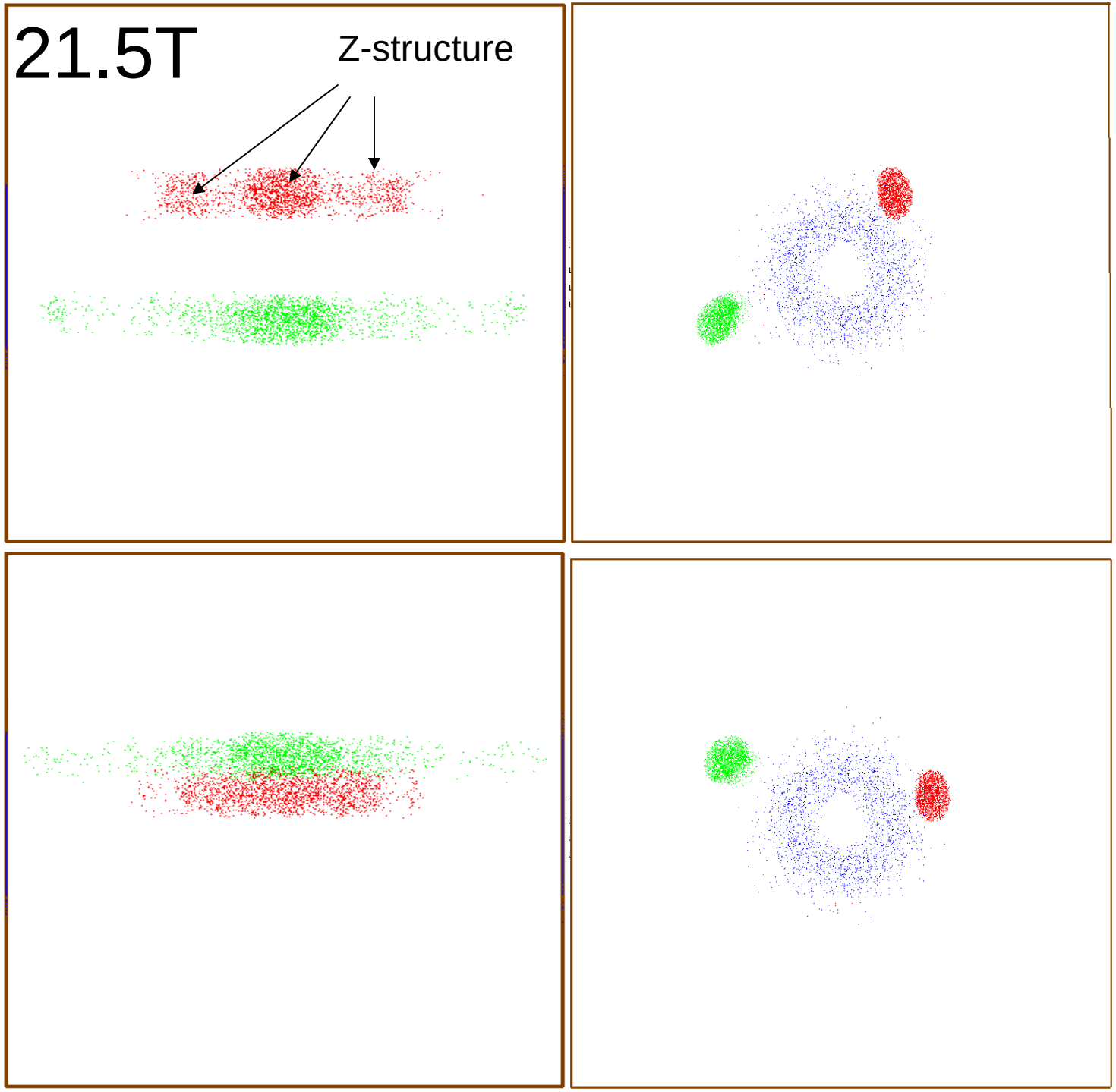
Axial structure formation



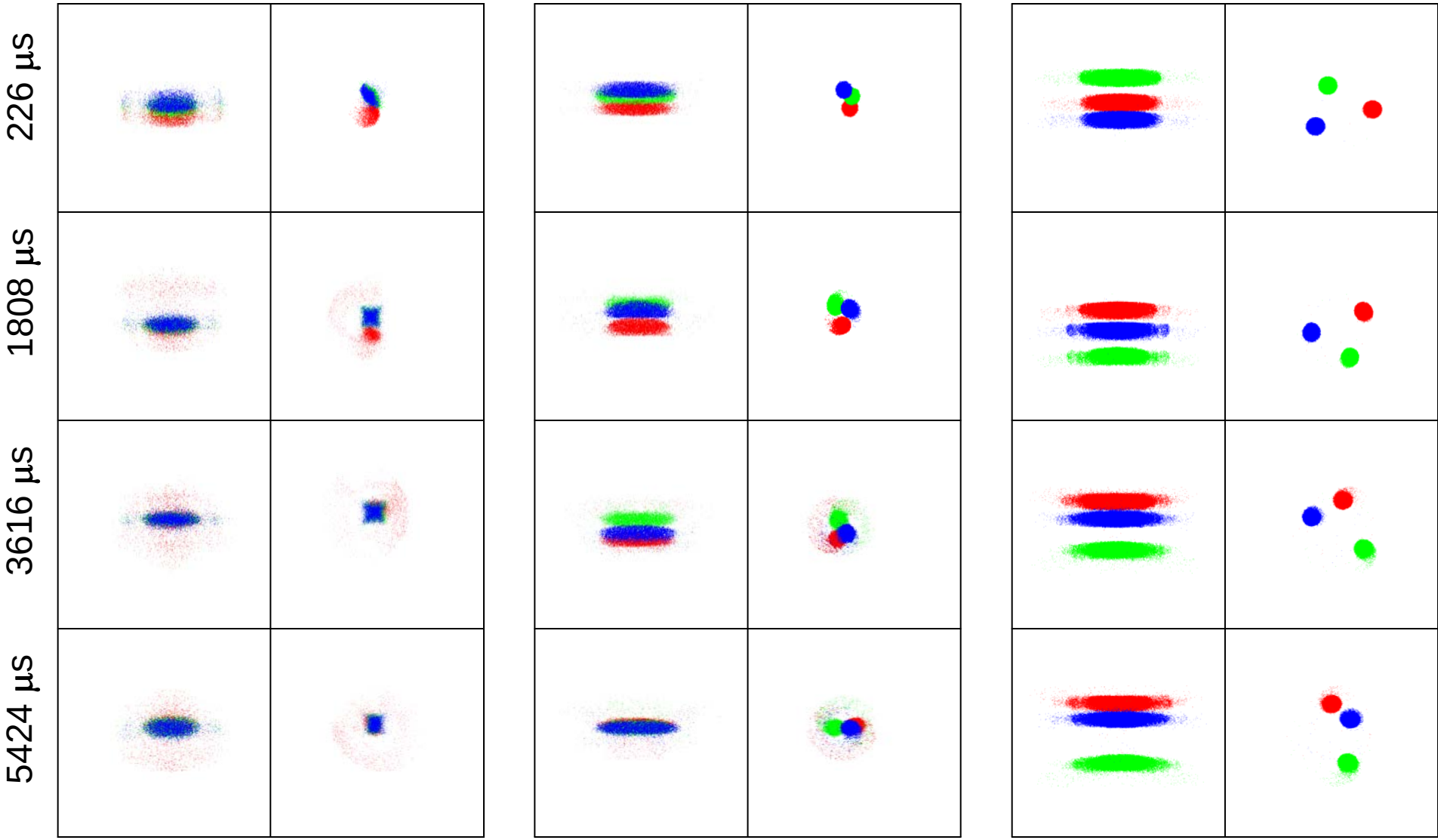
5



6


21.5T
Z-structure
34
50

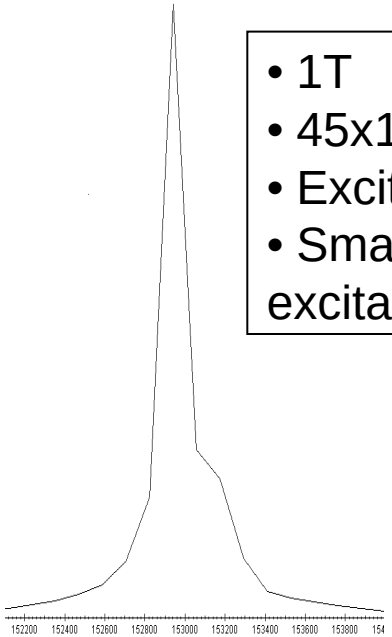
Dependence of coalescence on cyclotron excitation radius



1T 45x103 ions
Excitation 8.1 ms
Small radius of excitation

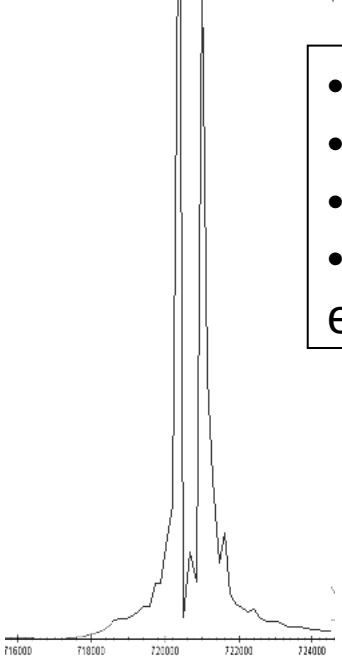
4.7T 225x103 ions
Excitation 37 ms
Small radius of excitation

4.7T 225x103 ions
Excitation 174 ms
Large radius of excitation



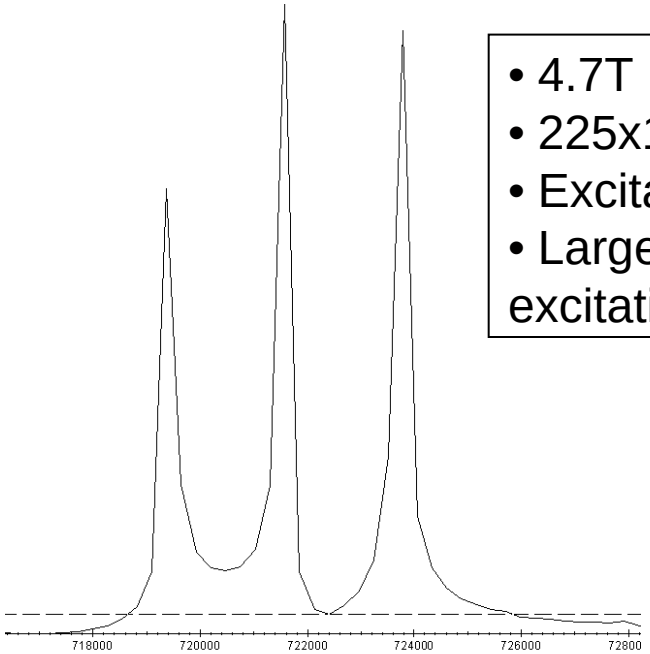
- 1T
- 45×10^3 ions
- Excitation 8.1 μs
- Small radius of excitation

Total coalescence



- 4.7T
- 225×10^3 ions
- Excitation 37 μs
- Small radius of excitation

Partial coalescence

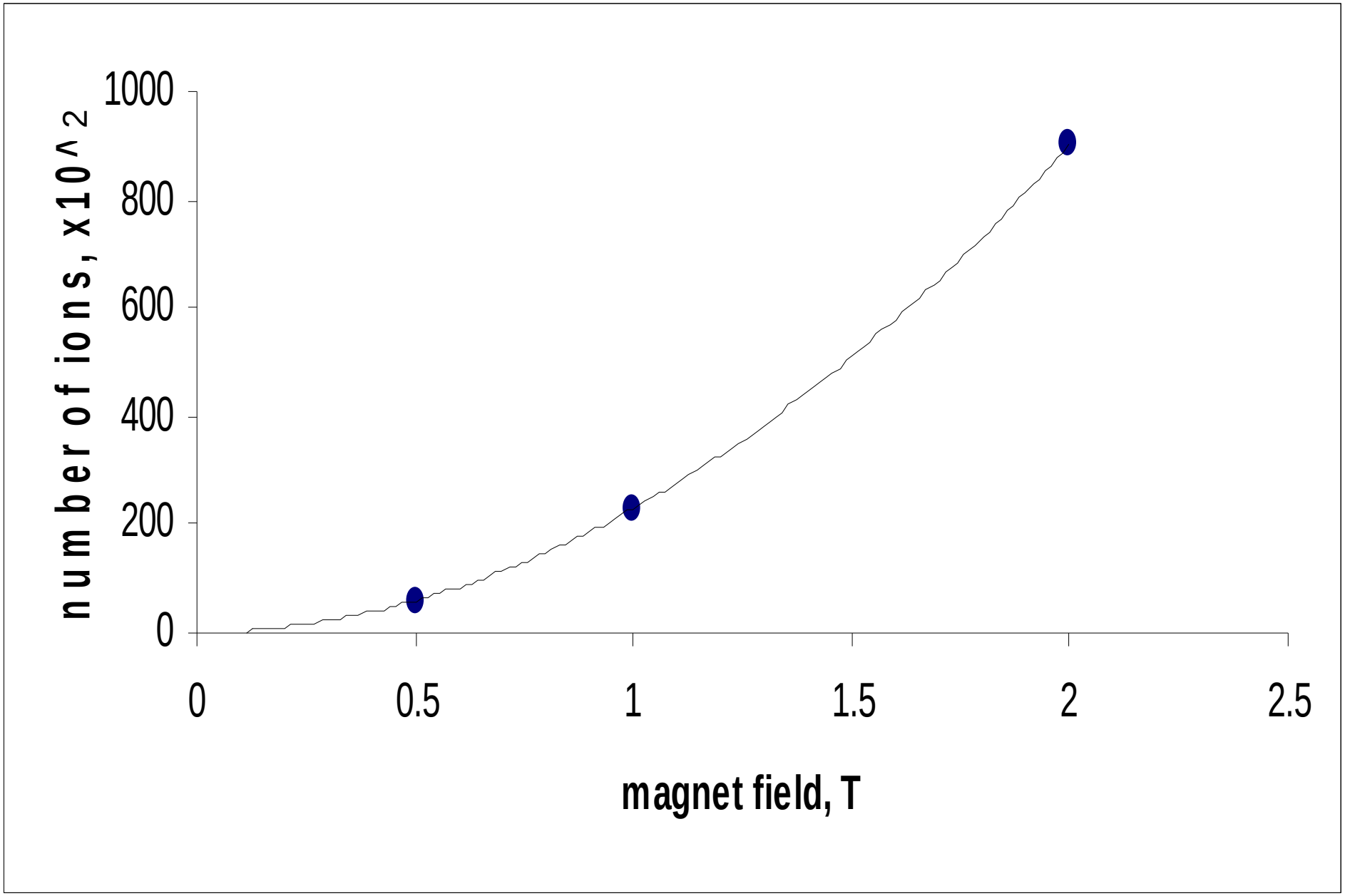


- 4.7T
- 225×10^3 ions
- Excitation 174 μs
- Large radius of excitation

No coalescence

Dependence of minimum number of ions

for coalescence to occur on magnetic field

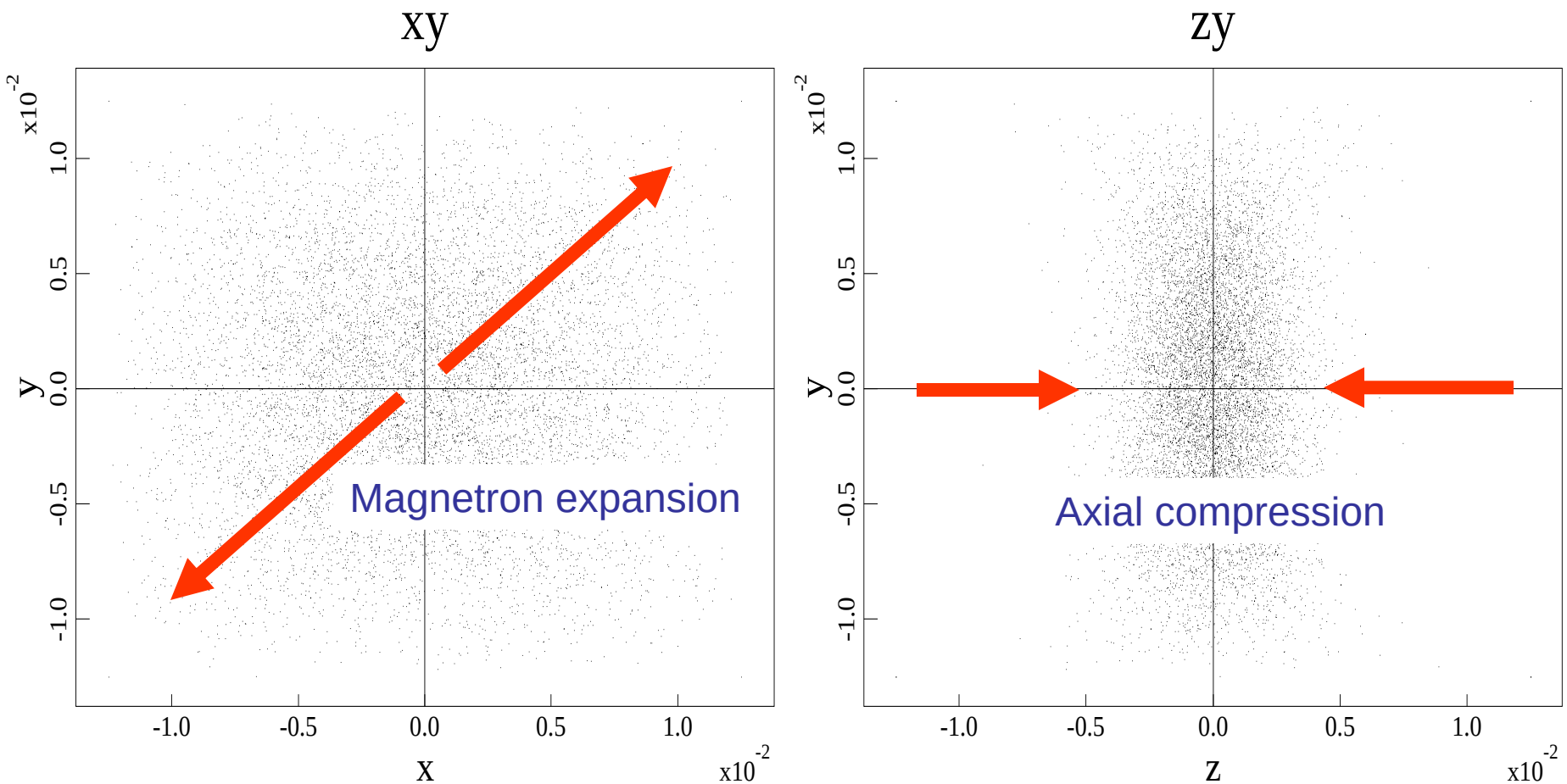

number of ions, x10^2
1000
800
600
400
200
0
0
0.5
1
1.5
2
2.5
magnet field, T

Initial ion cloud relaxation timing

Ion cloud evolution during relaxation period

P = 1.5 mtorr

Np = 25000

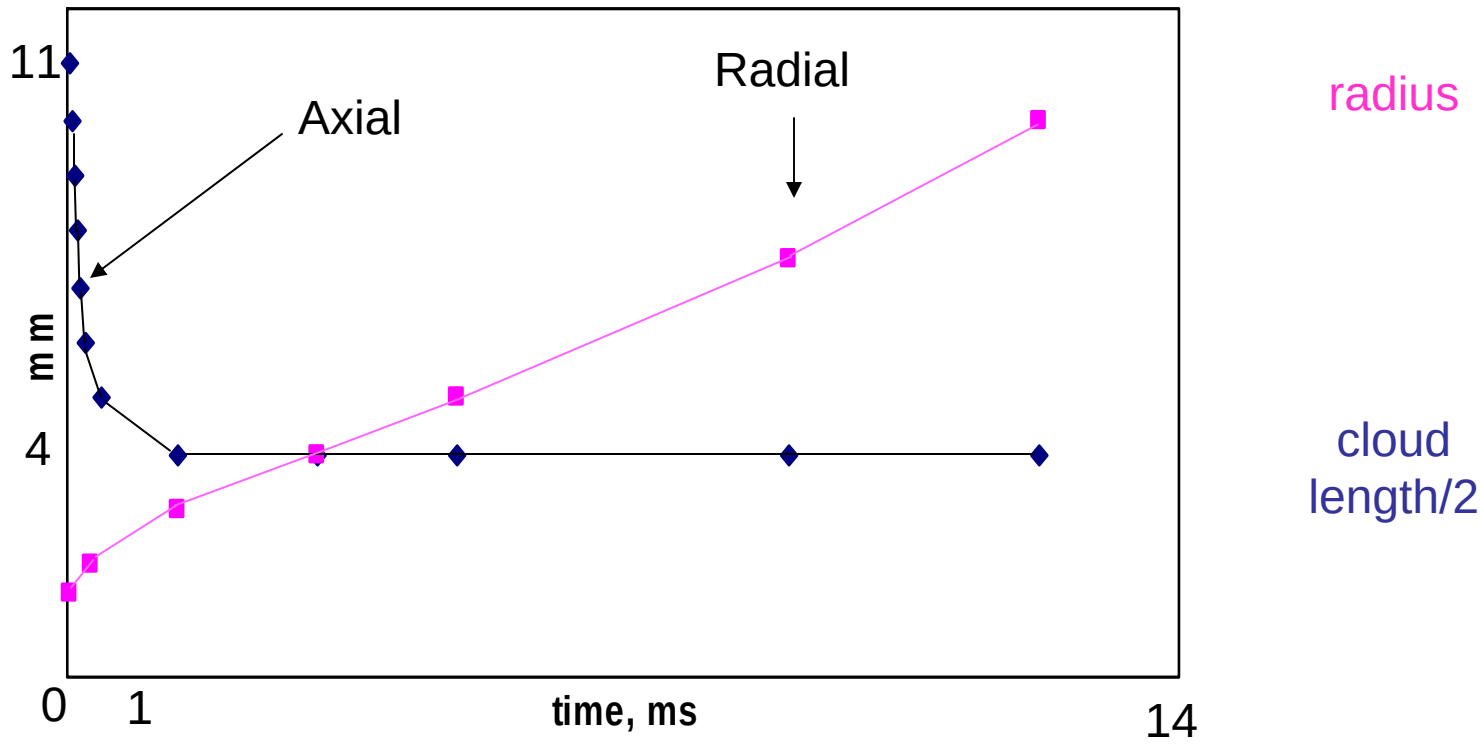


t = 18.2 ms

Ion cloud configuration during relaxation period

P = 1.5 mtorr

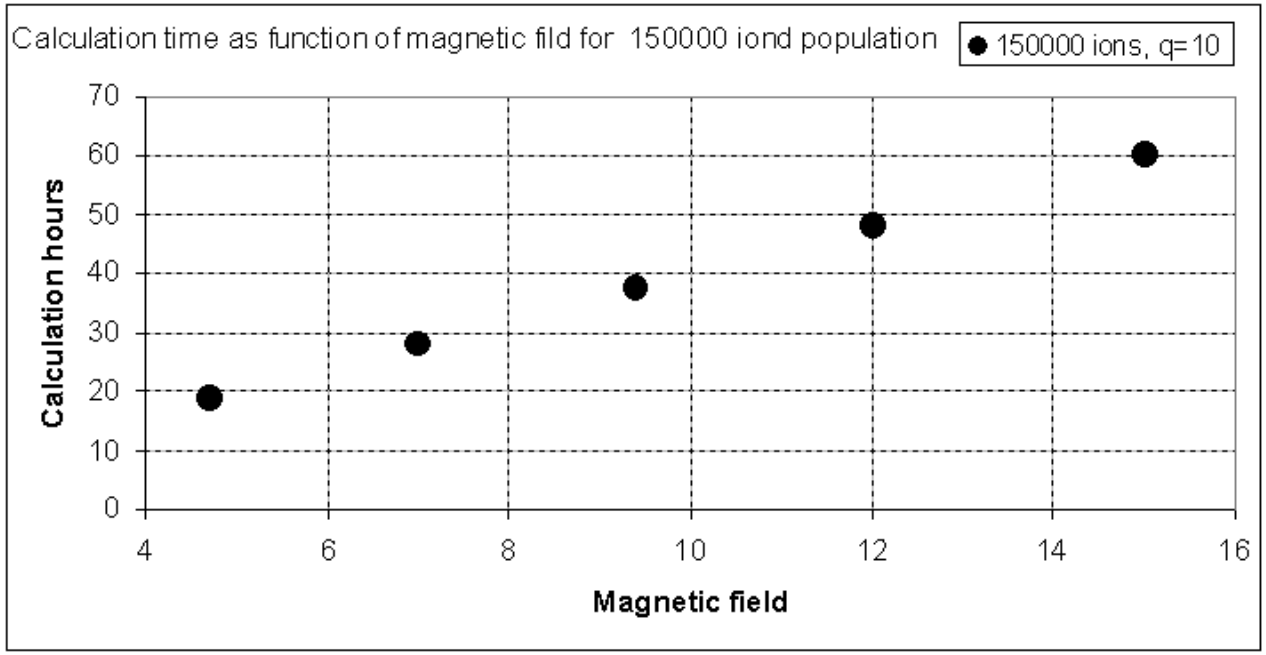
Np = 25000



Axial relaxation is 20 time faster than radial!!!

Calculation time

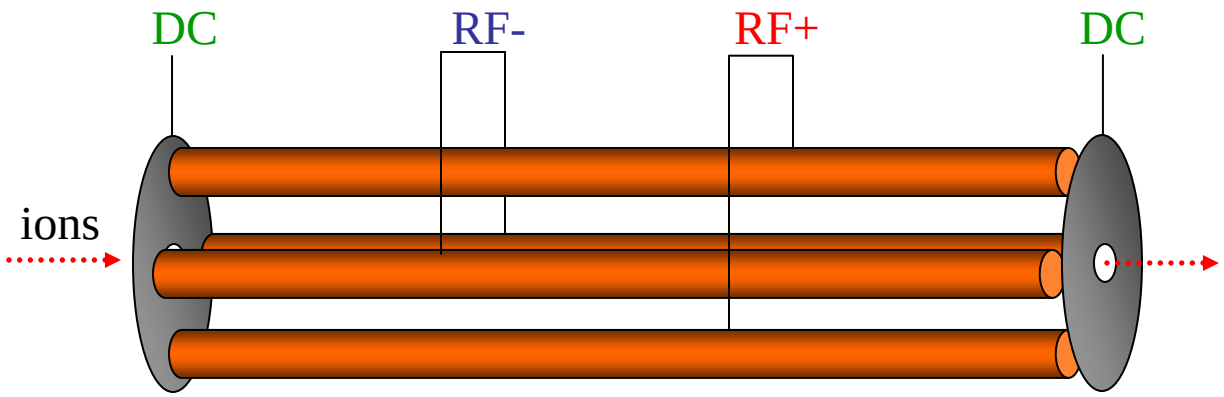
q=10 m=1000 Da N1=75000
q=10 m=1003 Da N2=75000
Totally 1500000 charges
Time domain duration=
0.0434 s



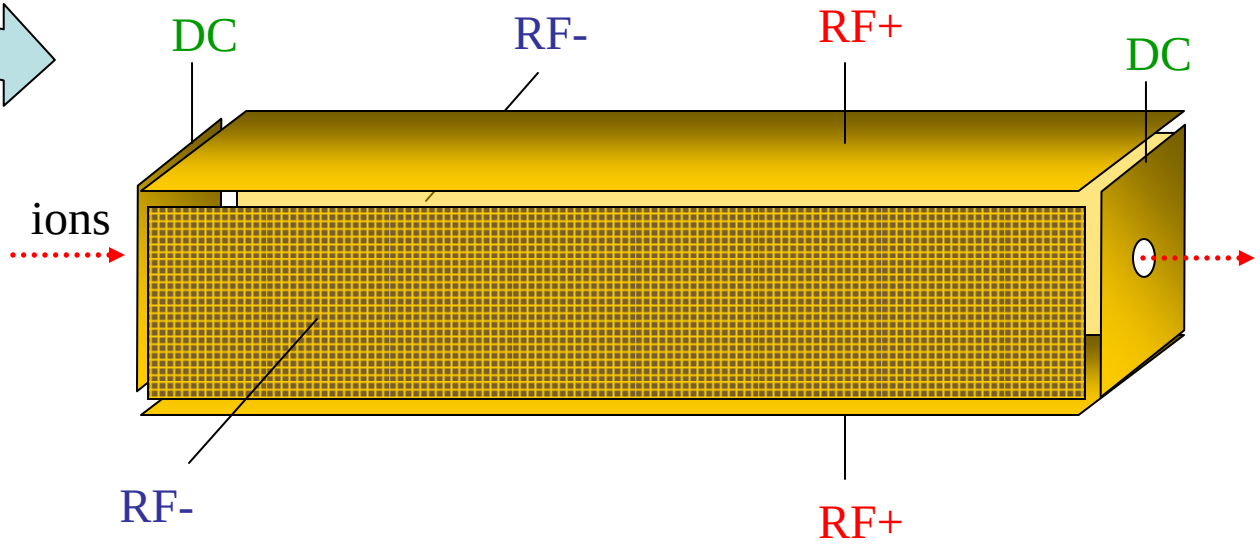
- **4.7 T** N1+N2 = 150000 Ndetc = 1848115 Nexct = 16435 T=0.0434 s Tcalc = 20 hours
- **7 T** N1+N2 = 150000 Ndetc = 2752512 Nexct = 36456 T=0.0434 s Tcalc = 30 hours
- **9.4 T** N1+N2 = 150000 Ndetc = 3696230 Nexct = 65739 T=0.0434 s Tcalc = 35 hours
- **12 T** N1+N2 = 150000 Ndetc = 4718592 Nexct = 107136 T=0.0434 s Tcalc = 54 hours
- **15 T** N1+N2 = 150000 Ndetc = 5898239 Nexct = 167397 T=0.0434 s Tcalc = 60 hours

Model of accumulation quadrupole

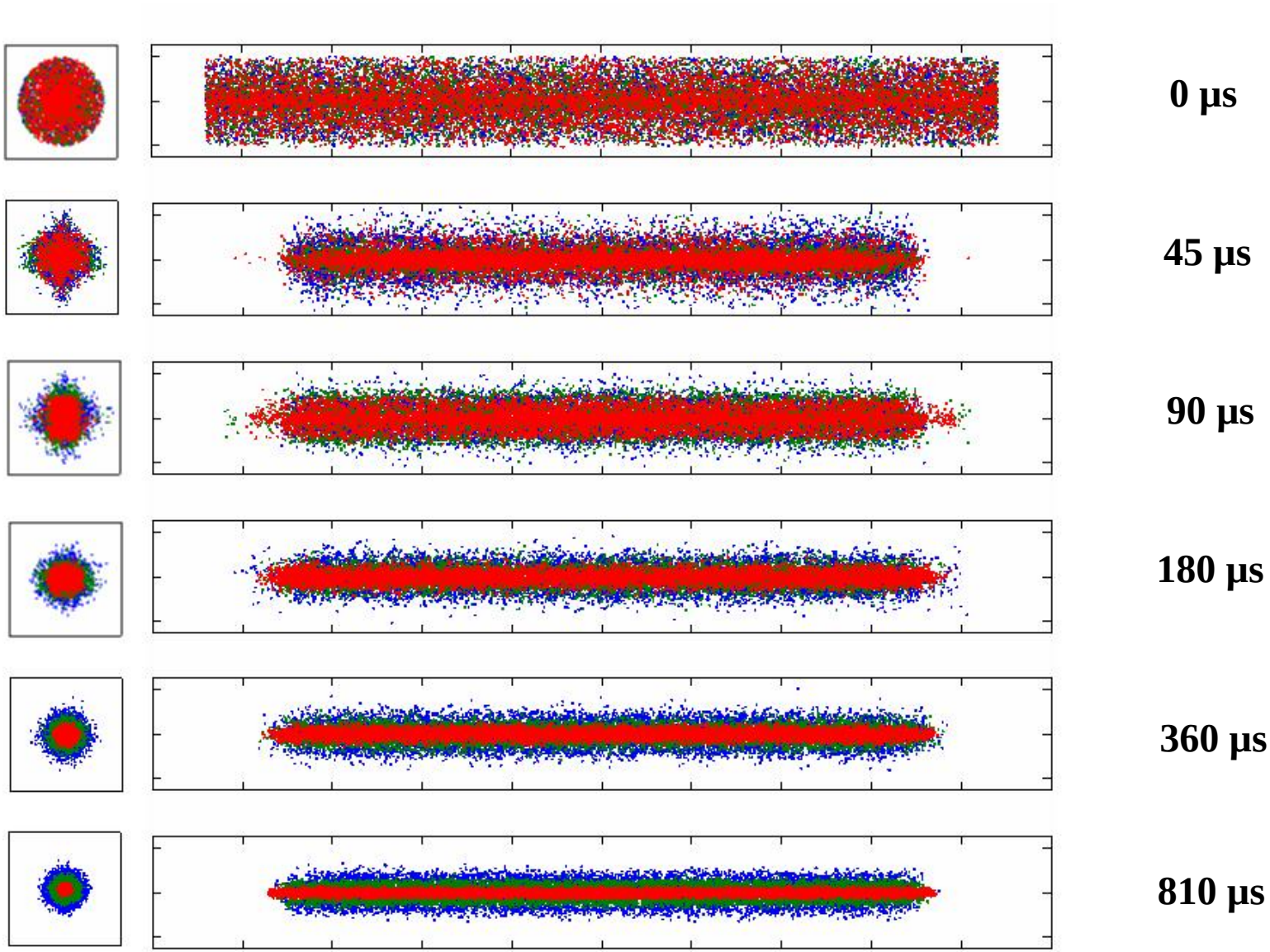
quadrupole scheme



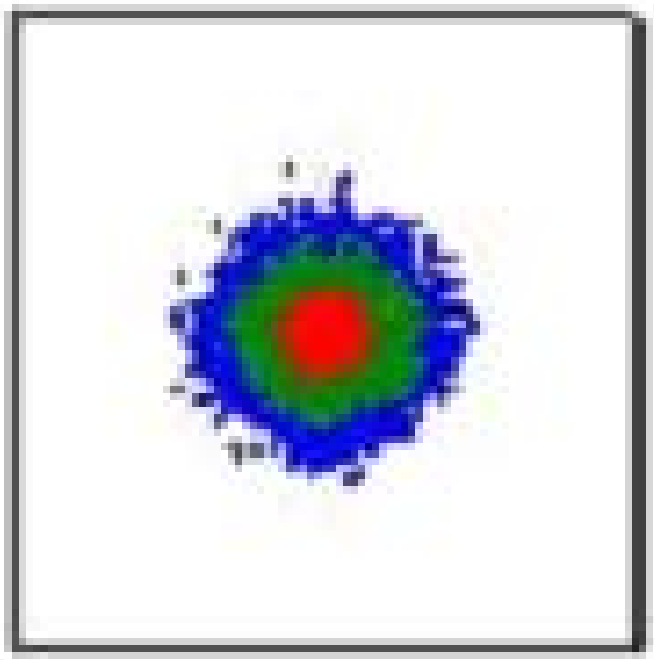
simulation model

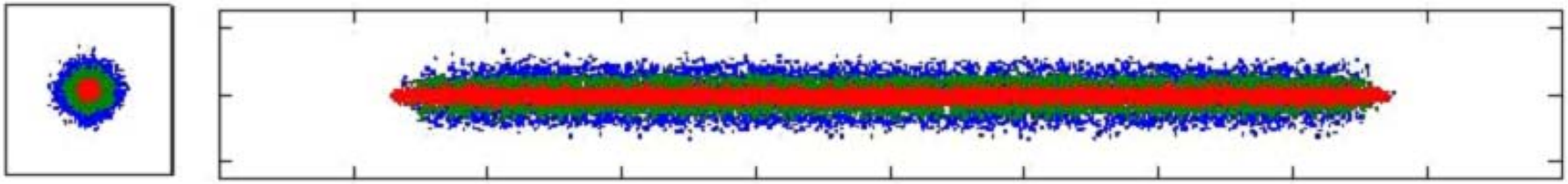


Accumulation quadrupole in a storage mode:3,000,000 ions totally of m/z 200(red), 500(green), 1000(blue) of equal amount.

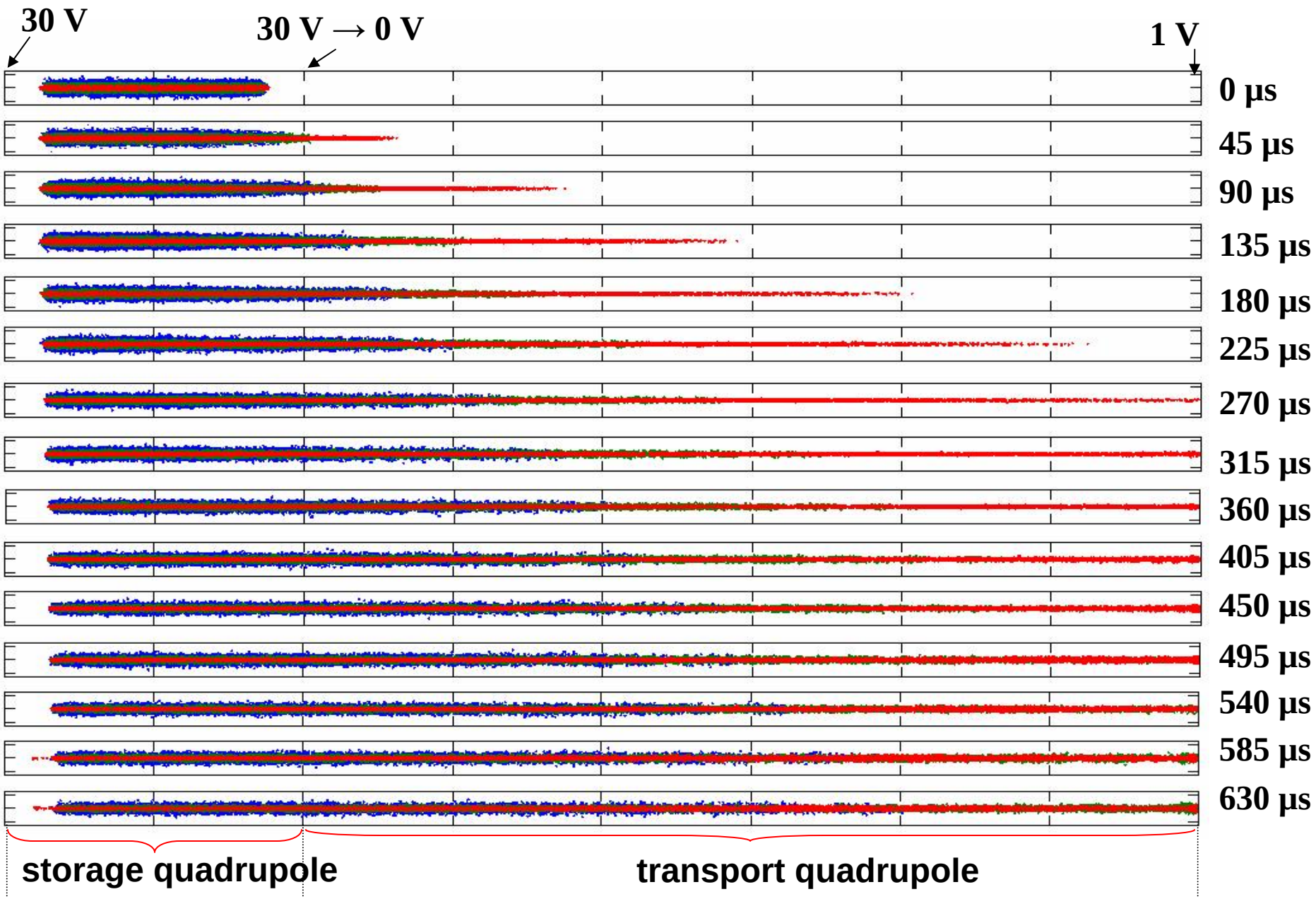


m/z 200(red), 500(green), 1000(blue) of equal amount

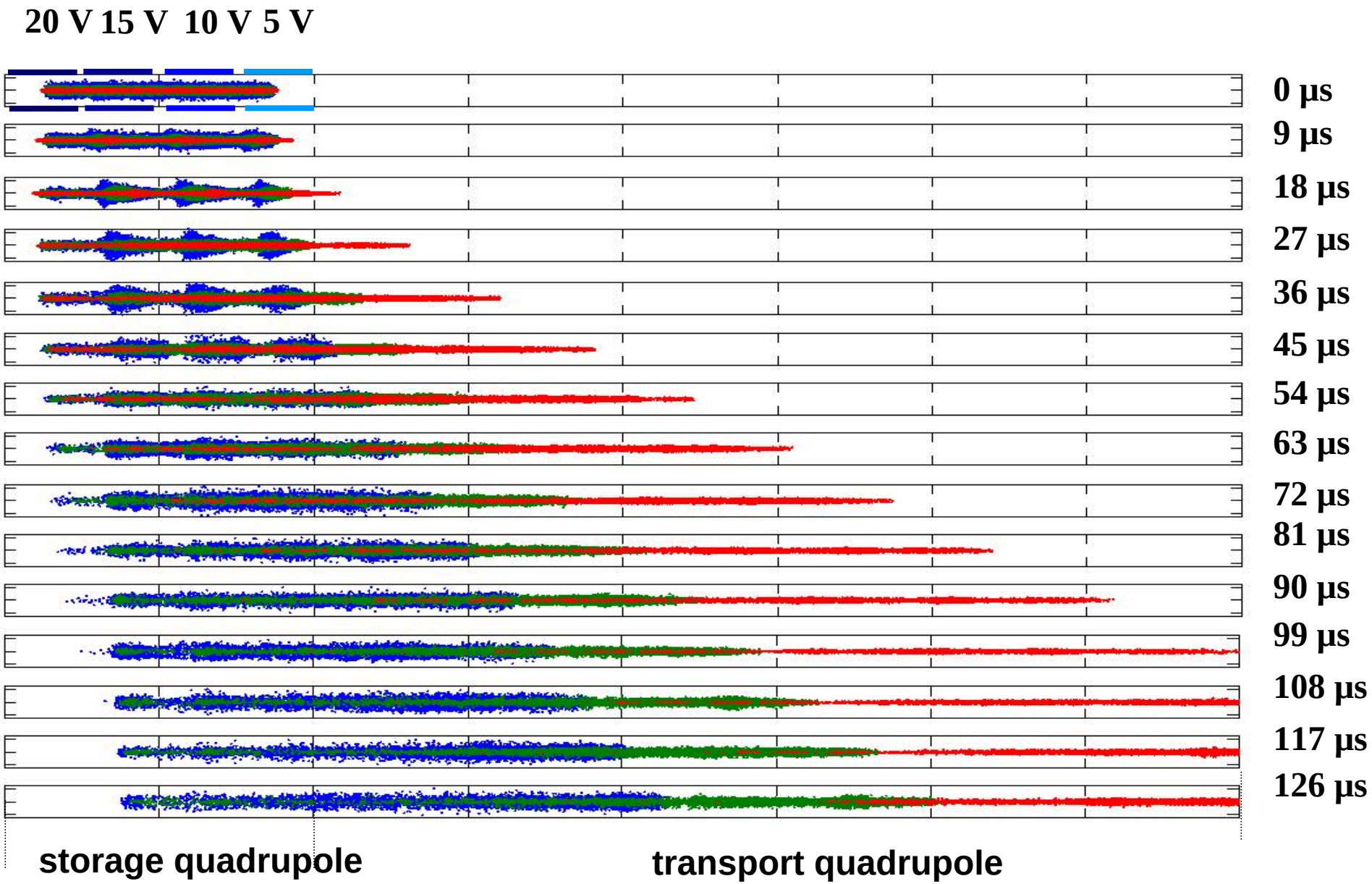




Behavior of ion cloud (3,000,000 ions) m/z 200(red), 500(green), 1000(blue), during its ejection from the accumulation quadrupole: nonsegmented accumulation quadrupole

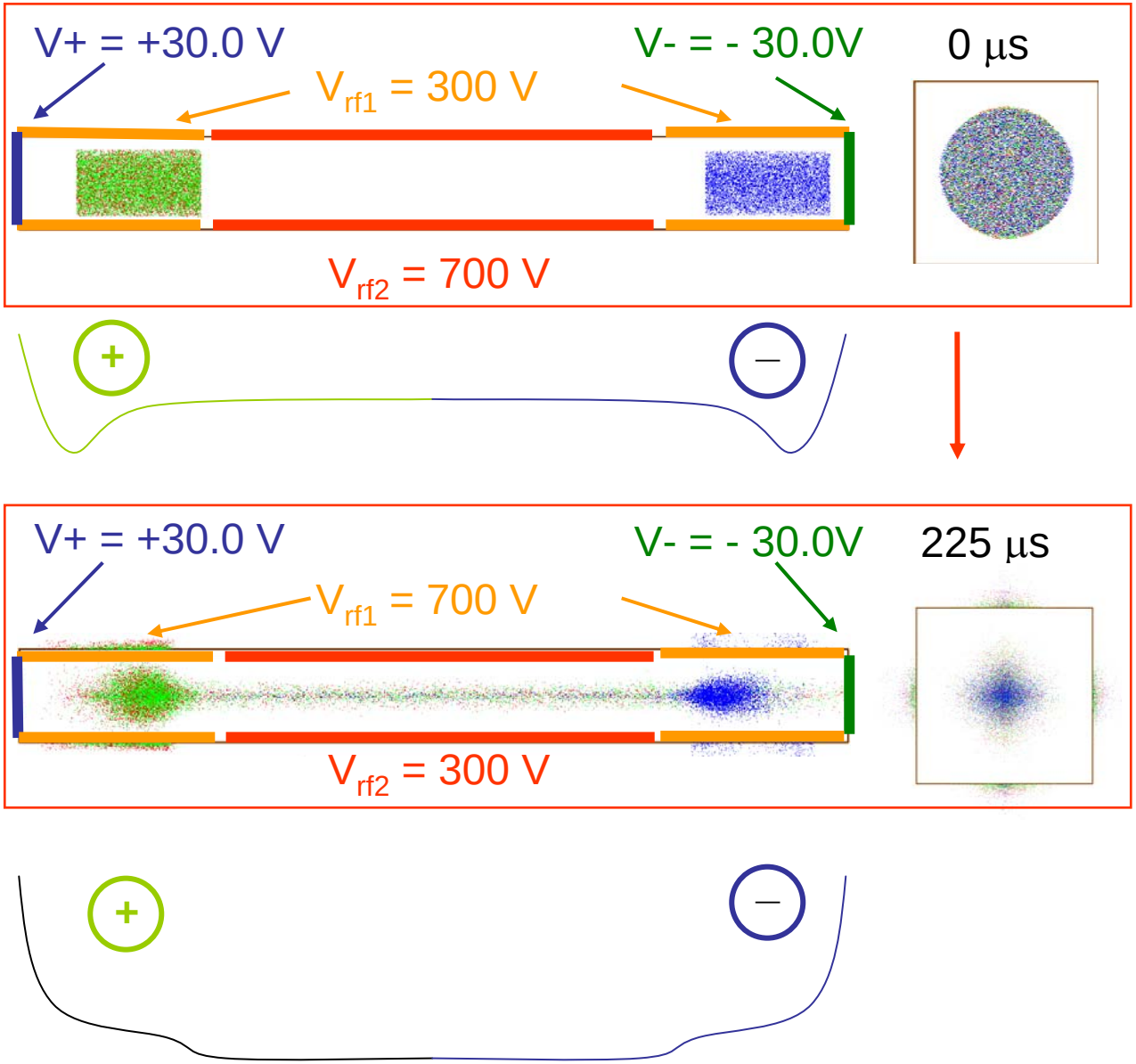


Behavior of ion cloud (3,000,000 ions) m/z 200(red), 500(green), 1000(blue), during its ejection from the accumulation quadrupole: segmented accumulation quadrupole



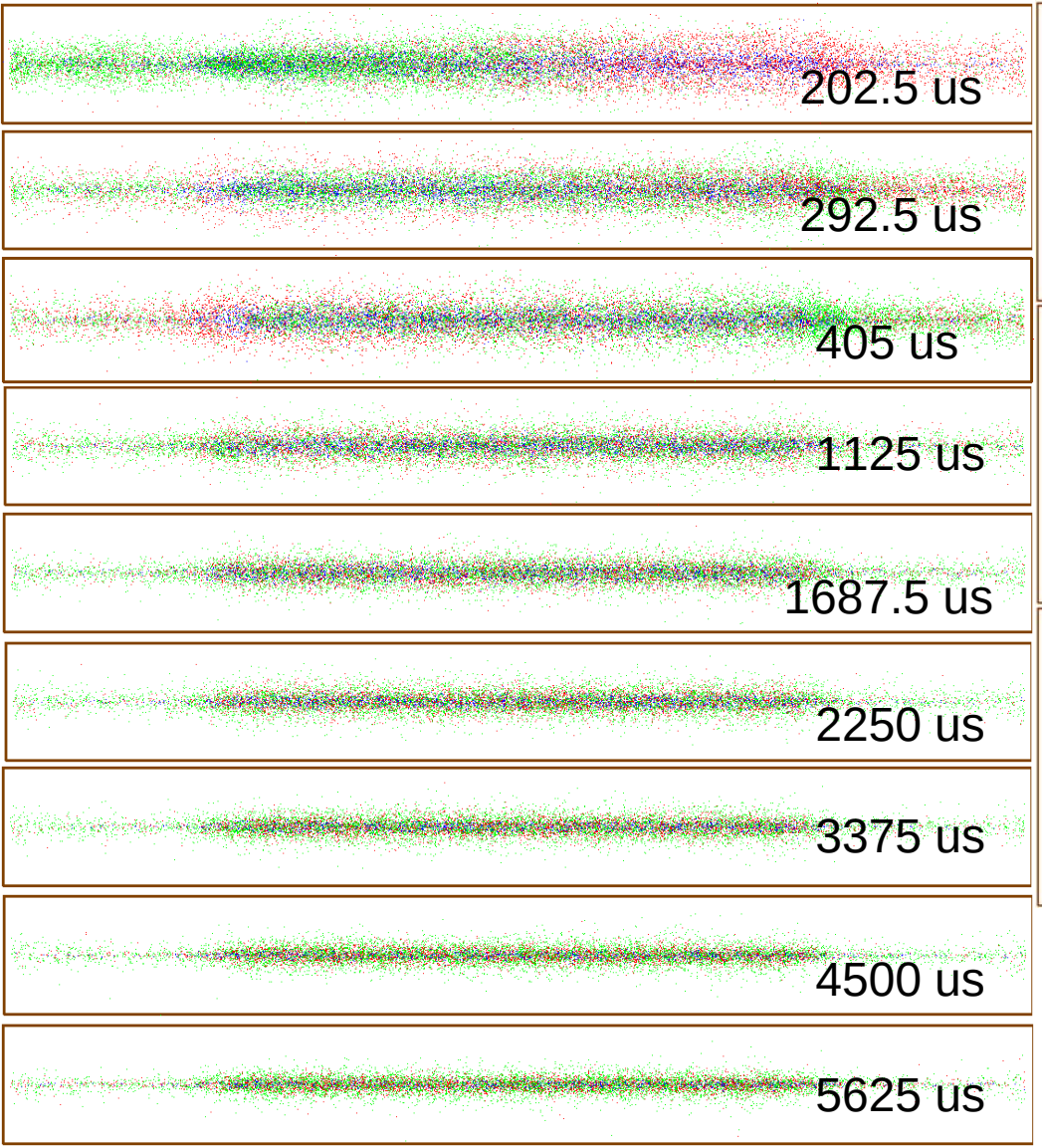
ETD in segmented quadrupols

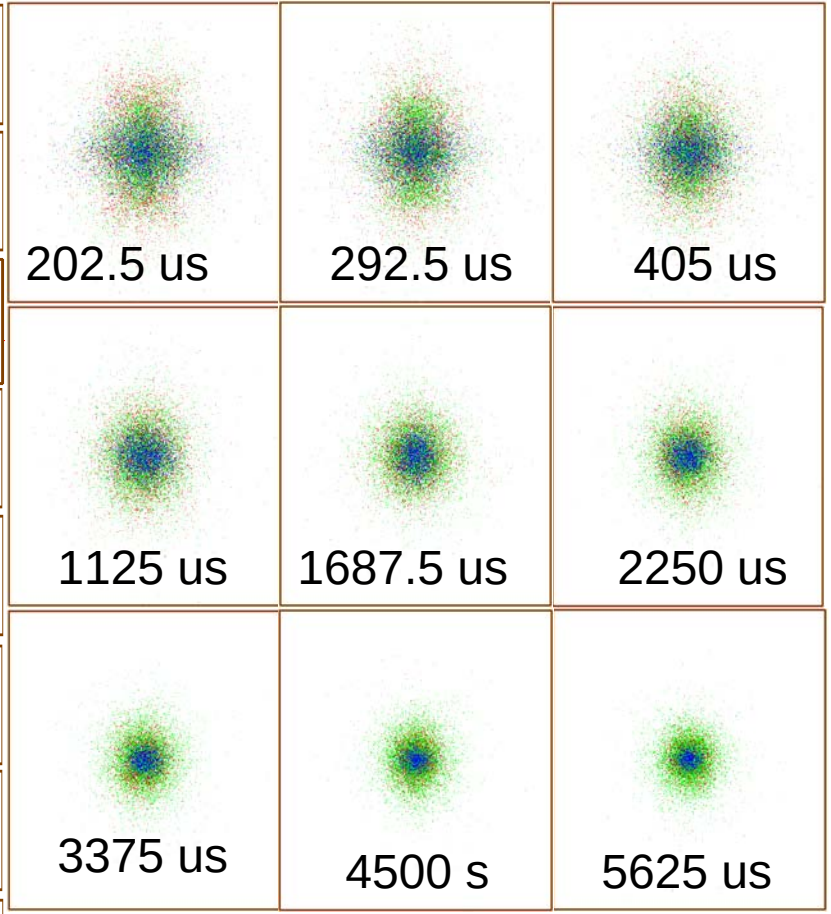
Scheme of ETD simulation in Linear segmented quadrupole.



Accumulation
2'500 steps
(225 μs)

ETD
50'000 steps
(4500 μs)



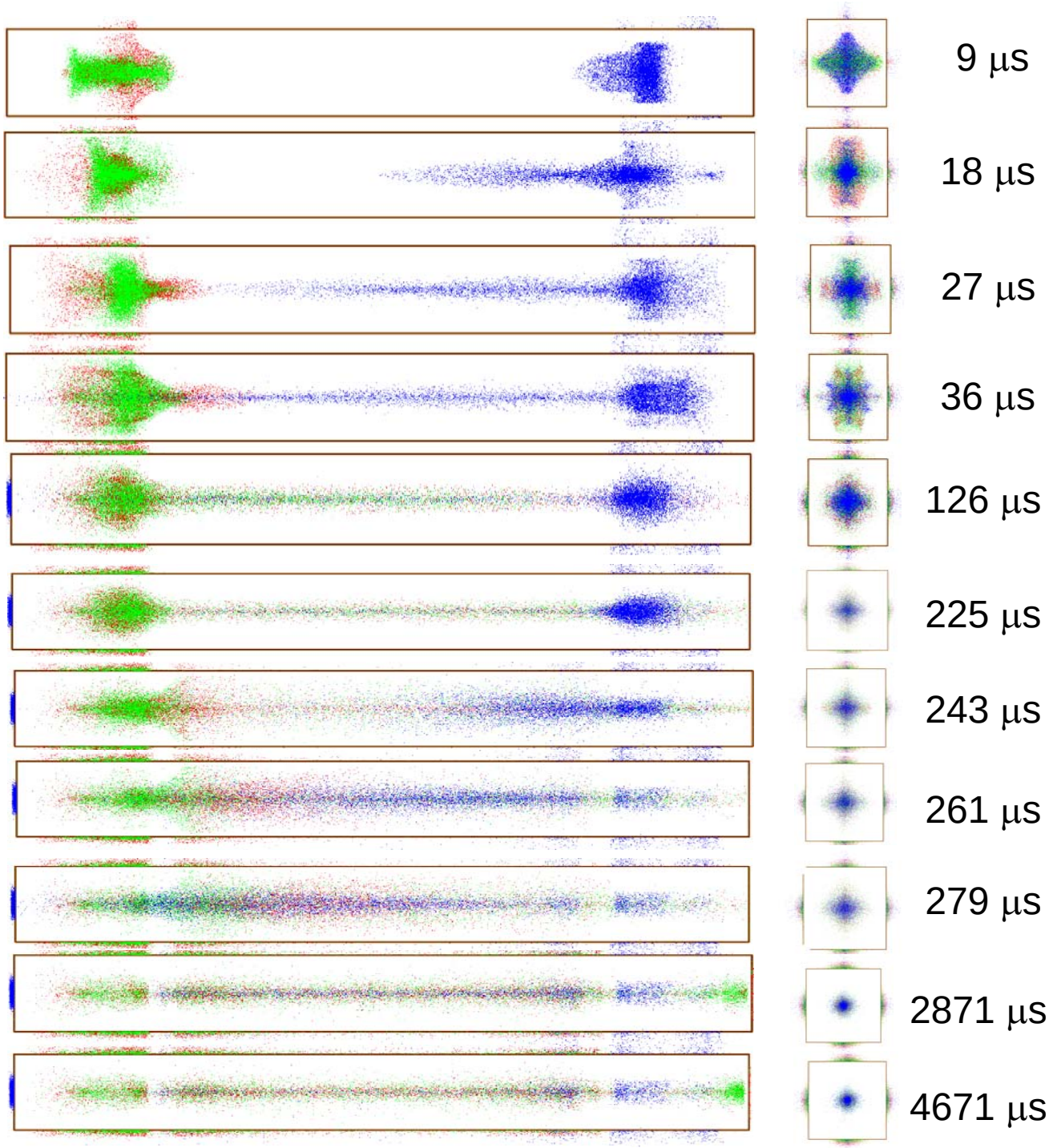


ν = 1100000 Hz

Initial conditions :

m1 = 500.0 q1 = 1.0 N1 = 1’000’000
m2 = 1000.0 q2 = 1.0 N2 = 1’000’000
m3 = 200.0 q3 = -1.0 N3 = 10’000’000

(**V**trap 0-p)=700,0 V
(**V**side,center 0-p)=150,0 V
(**V**side,ends 0-p)=350,0 V



Ion clouds dynamics in segmented quadrupole.

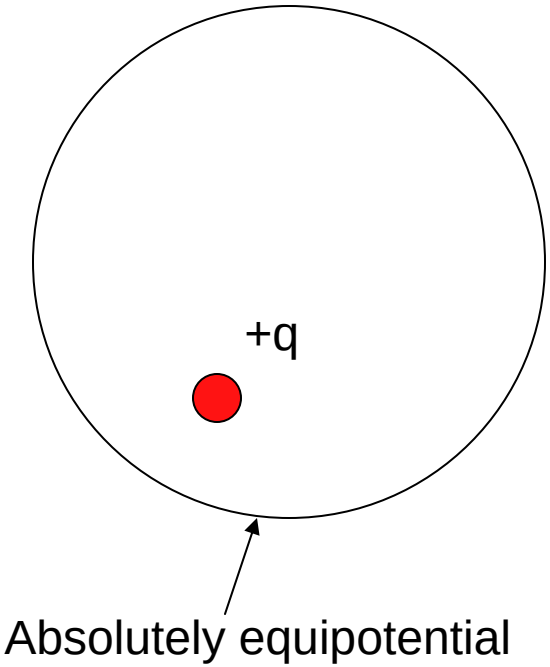
What was shown until now is made for rectangular geometry electrode assemblies

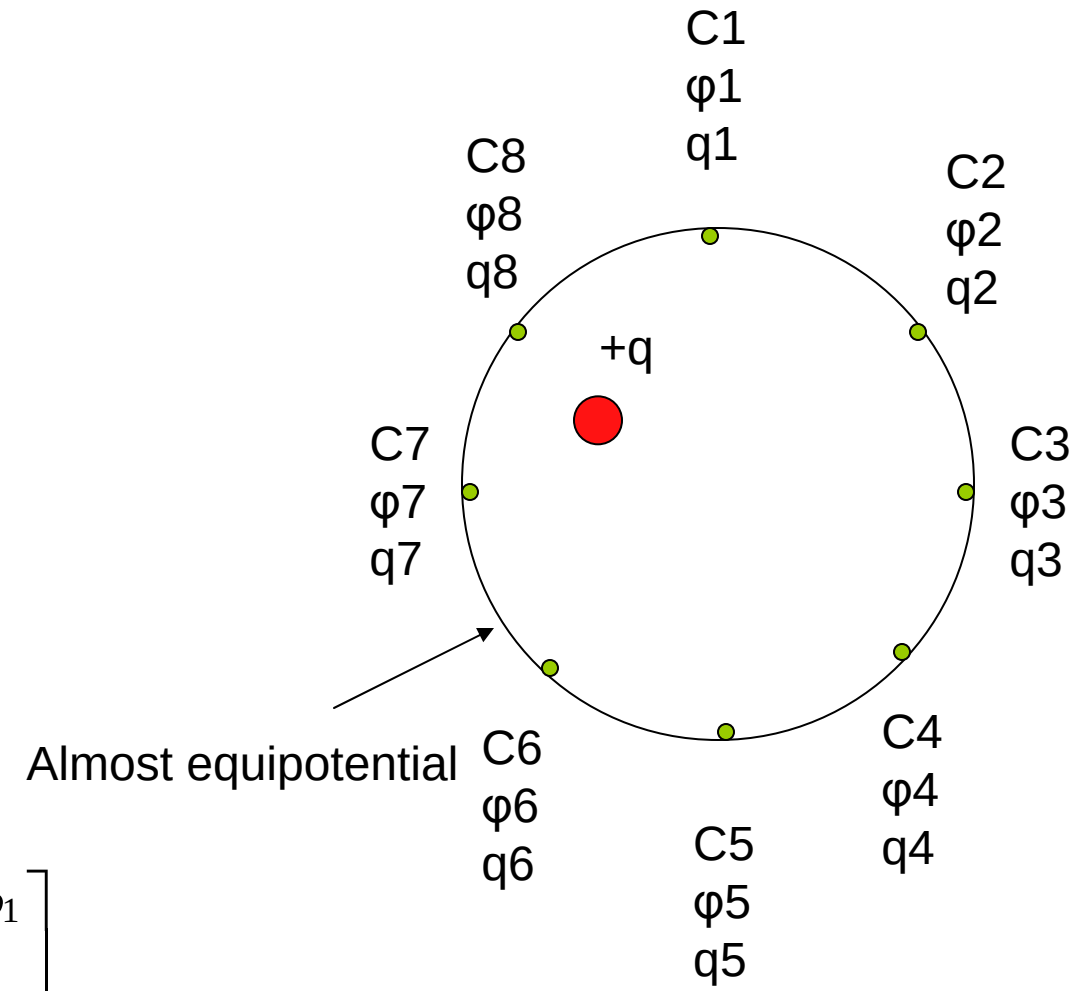
The goal

is to include arbitrary geometry electrodes

For arbitrary electrode forms Fast Fourier Transform method of Poisson equation solving is not applicable any more

Capacity matrix method



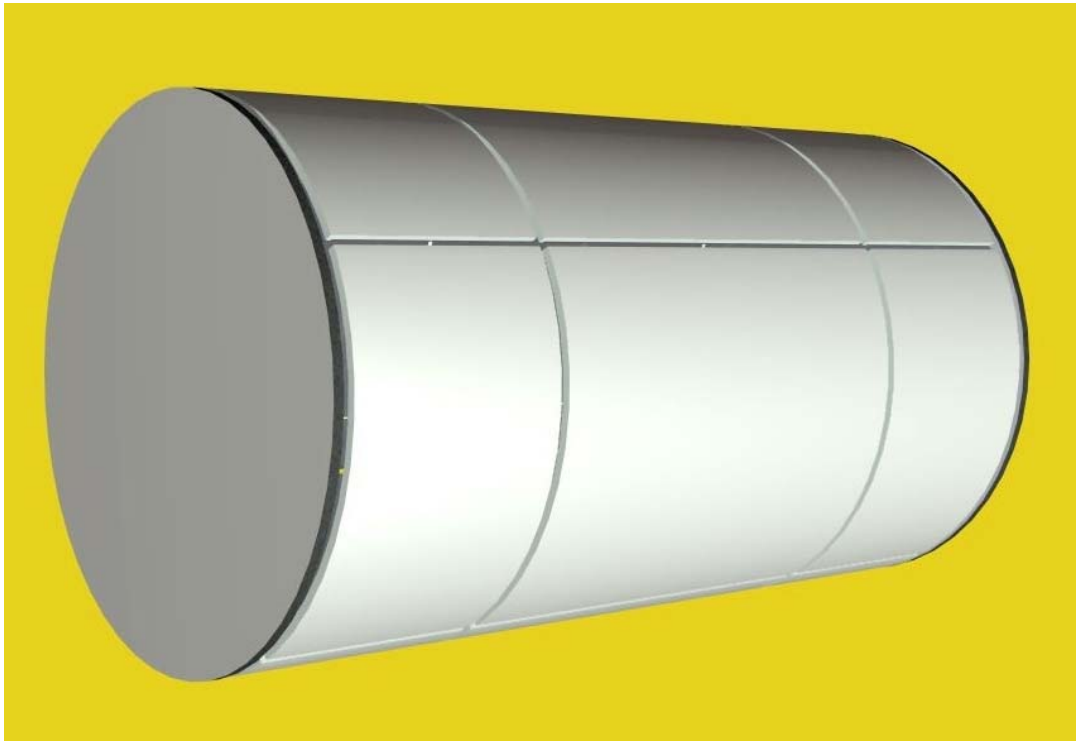


$$\begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} = \begin{pmatrix} & & \\ & C & \\ & & \end{pmatrix} \begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_n \end{bmatrix}$$

C is a capacity matrix

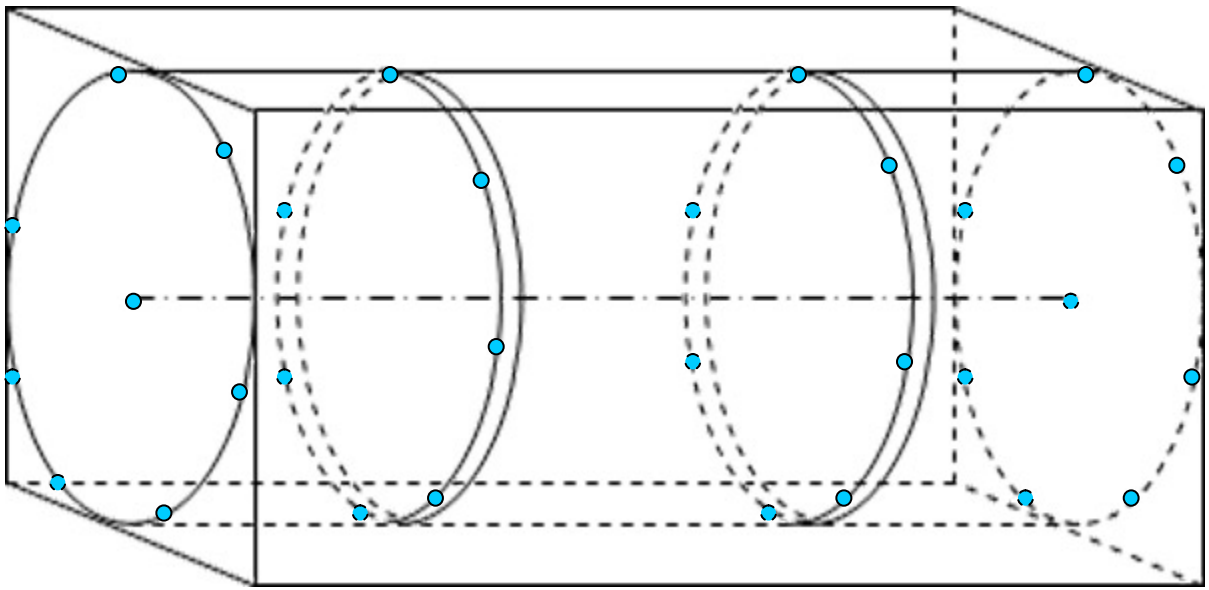
How to find capacity matrix?

Cylindrical ICR cell



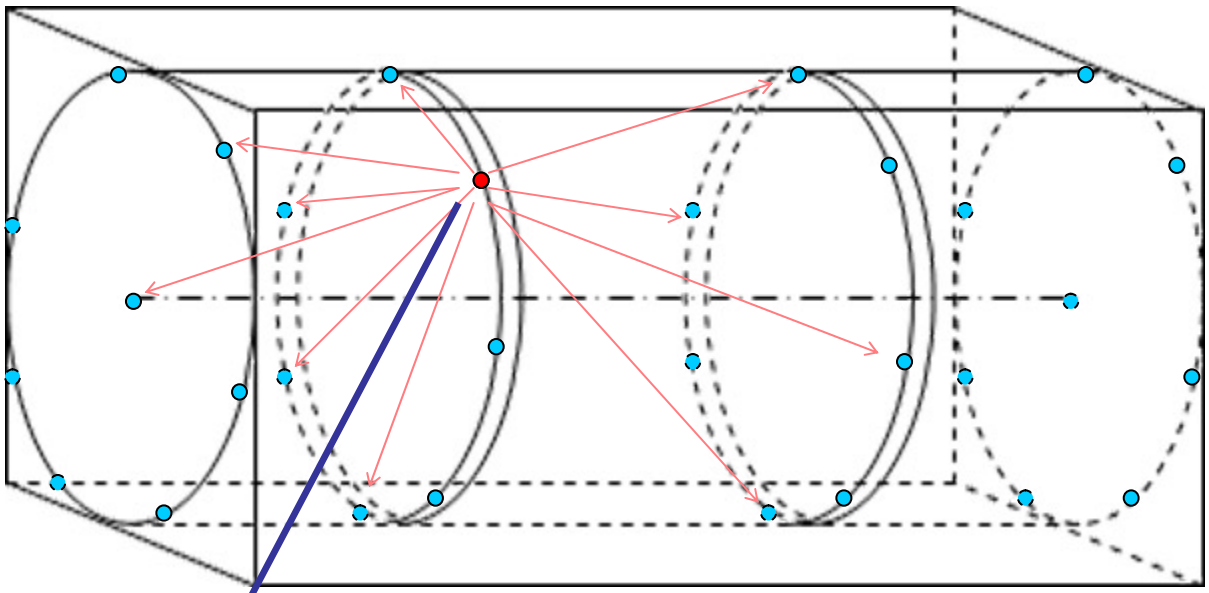
Capacity matrix method

Approximation of electrodes with nodes of computational grid



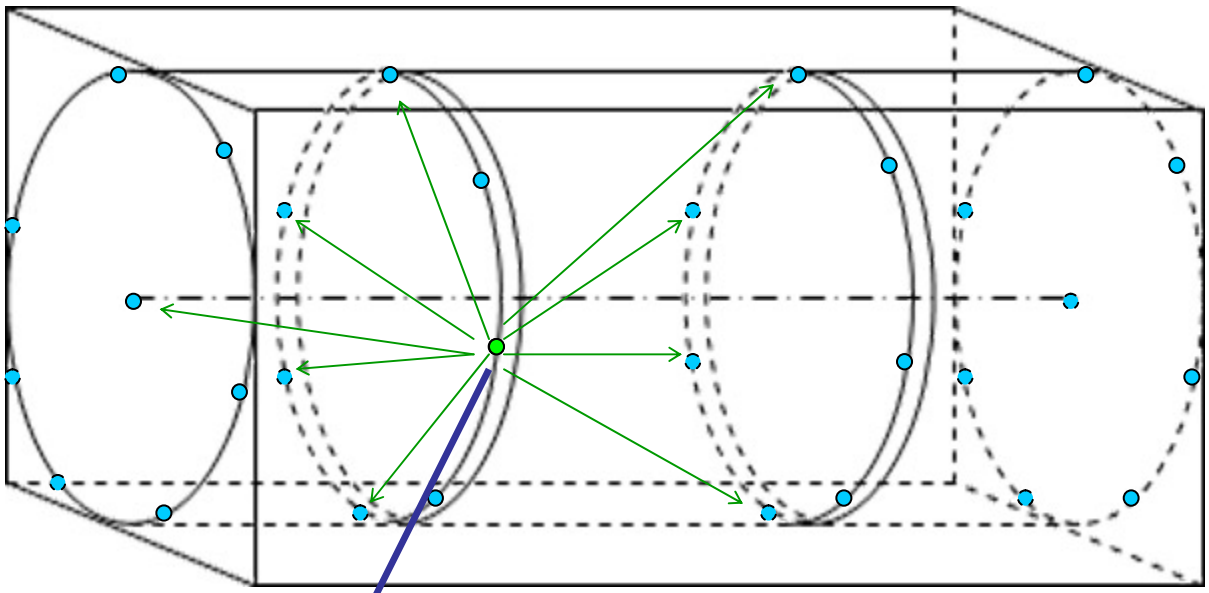
Electrodes have known potential. Our goal is to substitute them with point charges in such a way that potentials (on electrode surfaces) stayed the same.

Placing unit charge in each of these nodes and solving Poisson equation we find potentials in other nodes which form columns of matrix A (inverse capacity matrix as shown later)



$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{bmatrix} q_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_n \end{bmatrix}$$

Placing unit charge in each of these nodes and solving Poisson equation we find potentials in other nodes which form columns of matrix A (inverse capacity matrix as shown later)



$$\begin{pmatrix} a_{11} & \boxed{a_{12}} & \cdots & a_{1n} \\ \vdots & \boxed{\vdots} & \ddots & \vdots \\ a_{n1} & \boxed{a_{n2}} & \cdots & a_{nn} \end{pmatrix} \begin{bmatrix} 0 \\ q_2 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_n \end{bmatrix}$$

Multiplying both parts of equation by A^{-1} we come to a system of linear equations of a well known form (now you see why A^{-1} is called Capacity matrix)

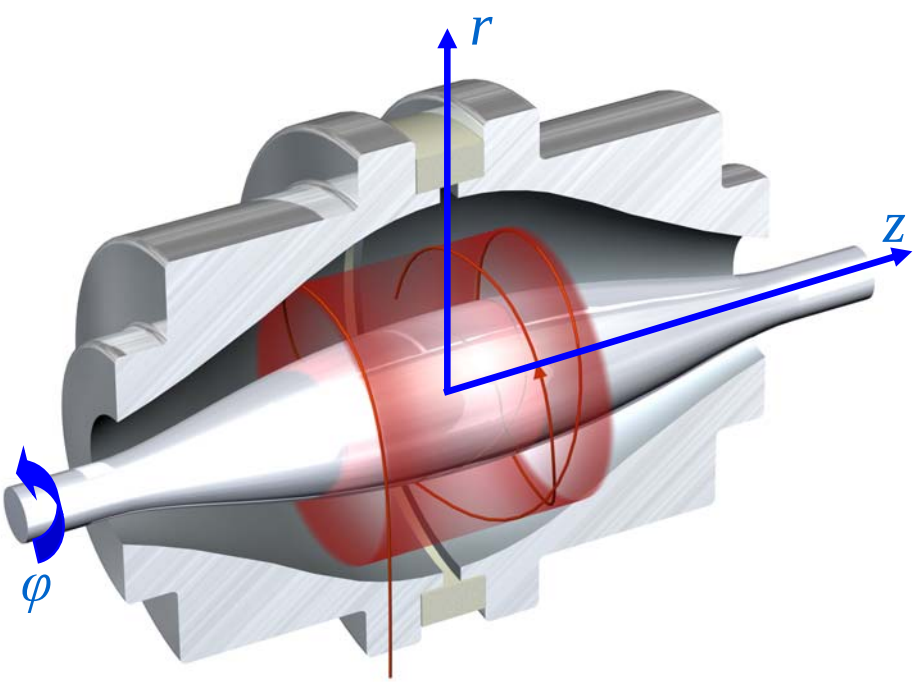
$$\begin{pmatrix} & & \\ & A & \\ & & \end{pmatrix}\begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} = \begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_n \end{bmatrix}$$

$$\begin{pmatrix} & & \\ & A^{-1} & \\ & & \end{pmatrix}\begin{pmatrix} & & \\ & A & \\ & & \end{pmatrix}\begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} = \begin{pmatrix} & & \\ & A^{-1} & \\ & & \end{pmatrix}\begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_n \end{bmatrix}$$

$$\begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} = \begin{pmatrix} & & \\ & C & \\ & & \end{pmatrix}\begin{bmatrix} \varphi_1 \\ \vdots \\ \varphi_n \end{bmatrix}$$

Capacity Matrix Method is applicable for arbitrary geometry electrodes.

Kingdon trap or Orbitrap



- Characteristic frequencies:
  - Frequency of rotation ω_φ
  - Frequency of radial oscillations ω_r
  - Frequency of axial oscillations ω_z

$$\omega_\varphi = \frac{\omega_z}{\sqrt{2}}\sqrt{\left(\frac{R_m}{R}\right)^2 - 1}$$

$$\omega_r = \omega_z\sqrt{\left(\frac{R_m}{R}\right)^2 - 2}$$

$$\omega_z = \sqrt{\frac{k}{m/q}}$$

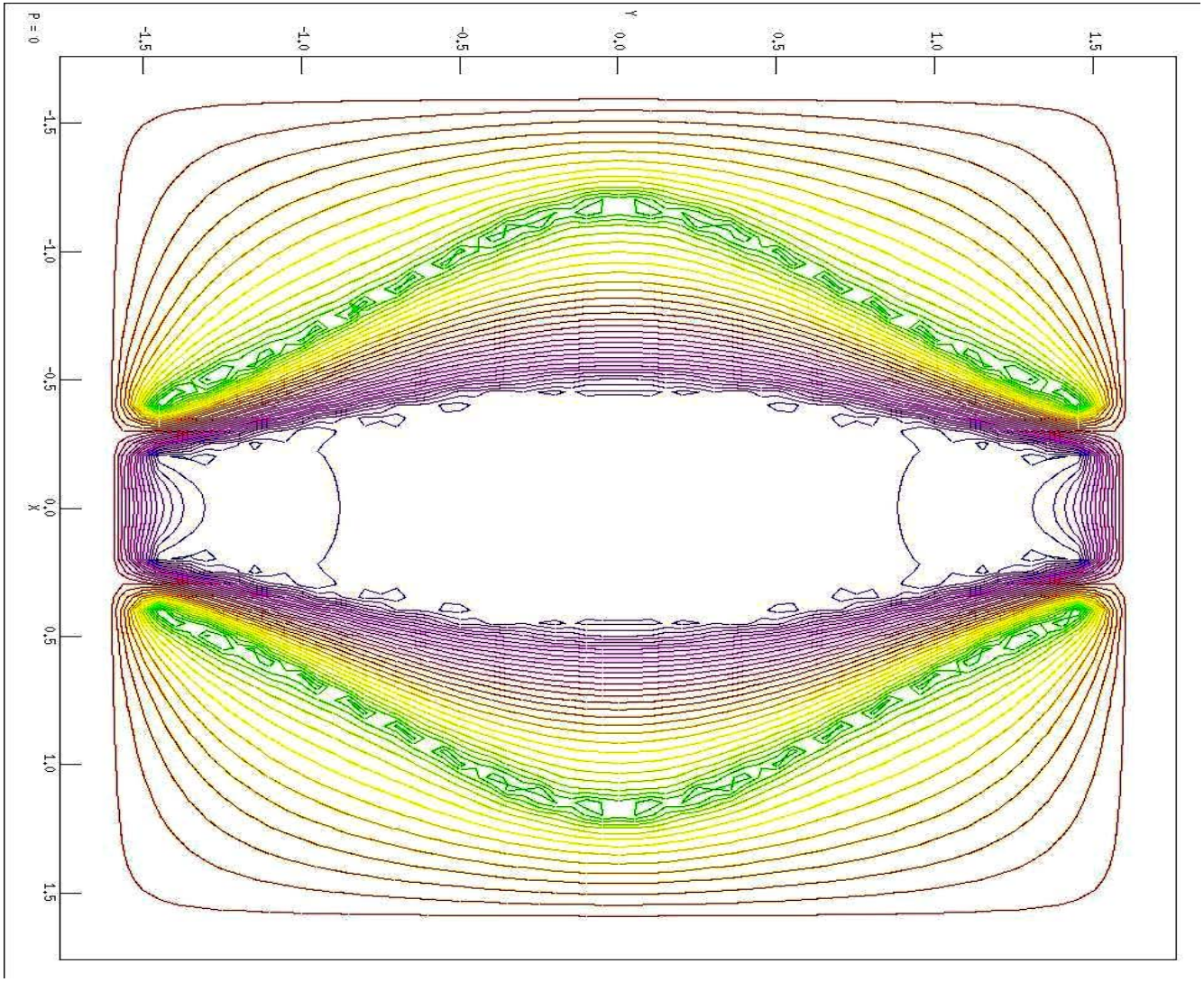
Only this frequency does not depend on energy, angle, etc. and could be used for mass analysis !

Hyper-logarithmic potential distribution in the Orbitrap: "ideal Kingdon trap"

$$U(r,z) = \frac{k}{2}\cdot\left\{z^2 - r^2/2 + R_m^2\cdot\ln(r/R_m)\right\}$$

•Korsunskii M.I., Basakutsa V.A. *Sov. Physics-Tech. Phys.* 1958; **3**: 1396.
•Knight R.D. *Appl.Phys.Lett.* 1981, **38**: 221.
•Gall L.N.,Golikov Y.K.,Aleksandrov M.L.,Pechalina Y.E.,Holin N.A. *SU Pat. 1247973*, 1986.

Slide from Alexander Makarov



Electric field distribution calculated by capacity matrix method for Kingdon trap

The entire algorithm for calculating the field acting on an ion, executed at each time step

Calculating matrix C
(done once before the main loop)

⇩

Solving Poisson equation in absence of electrodes

⇩

Multiplication of potentials in electrode nodes by C matrix
to calculate charges that imitate the electrodes

⇩

Solving Poisson equation to get real potential distribution

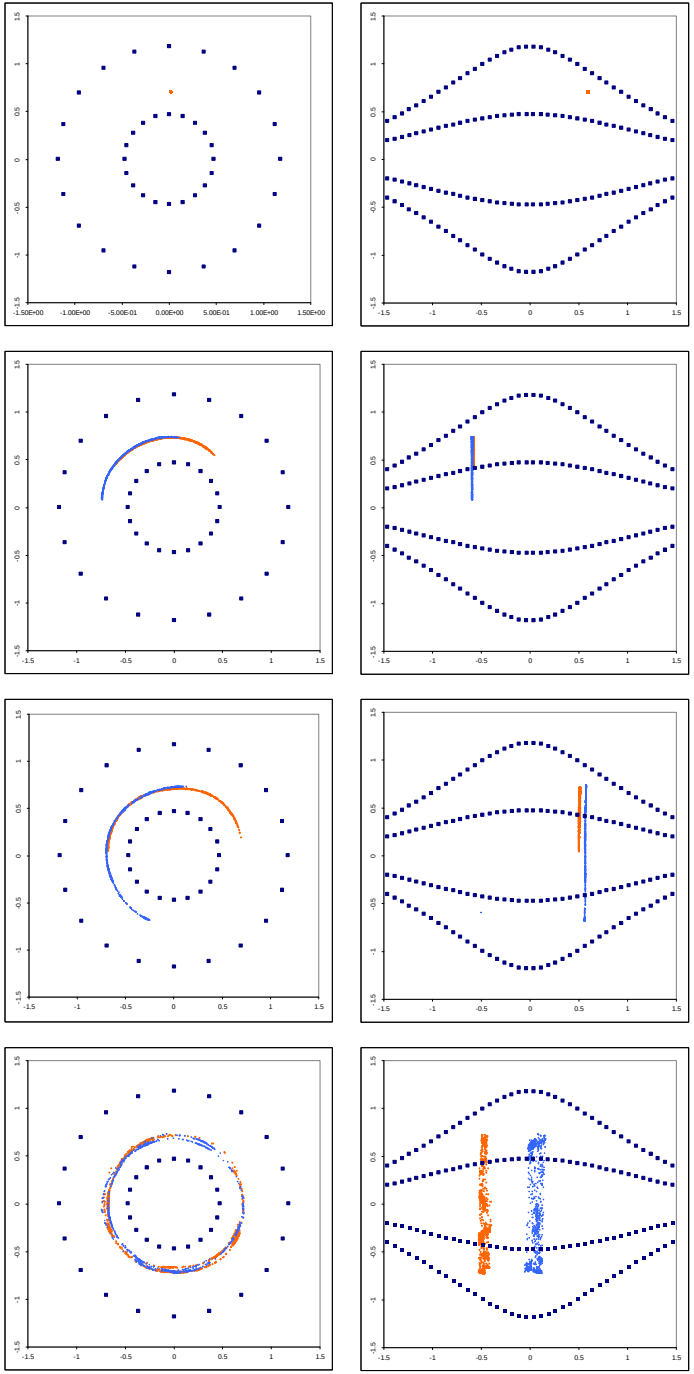
Evolution of ion clouds

In an Orbitrap

1000 charges

Blue 100 Da

Orange 100.5 Da

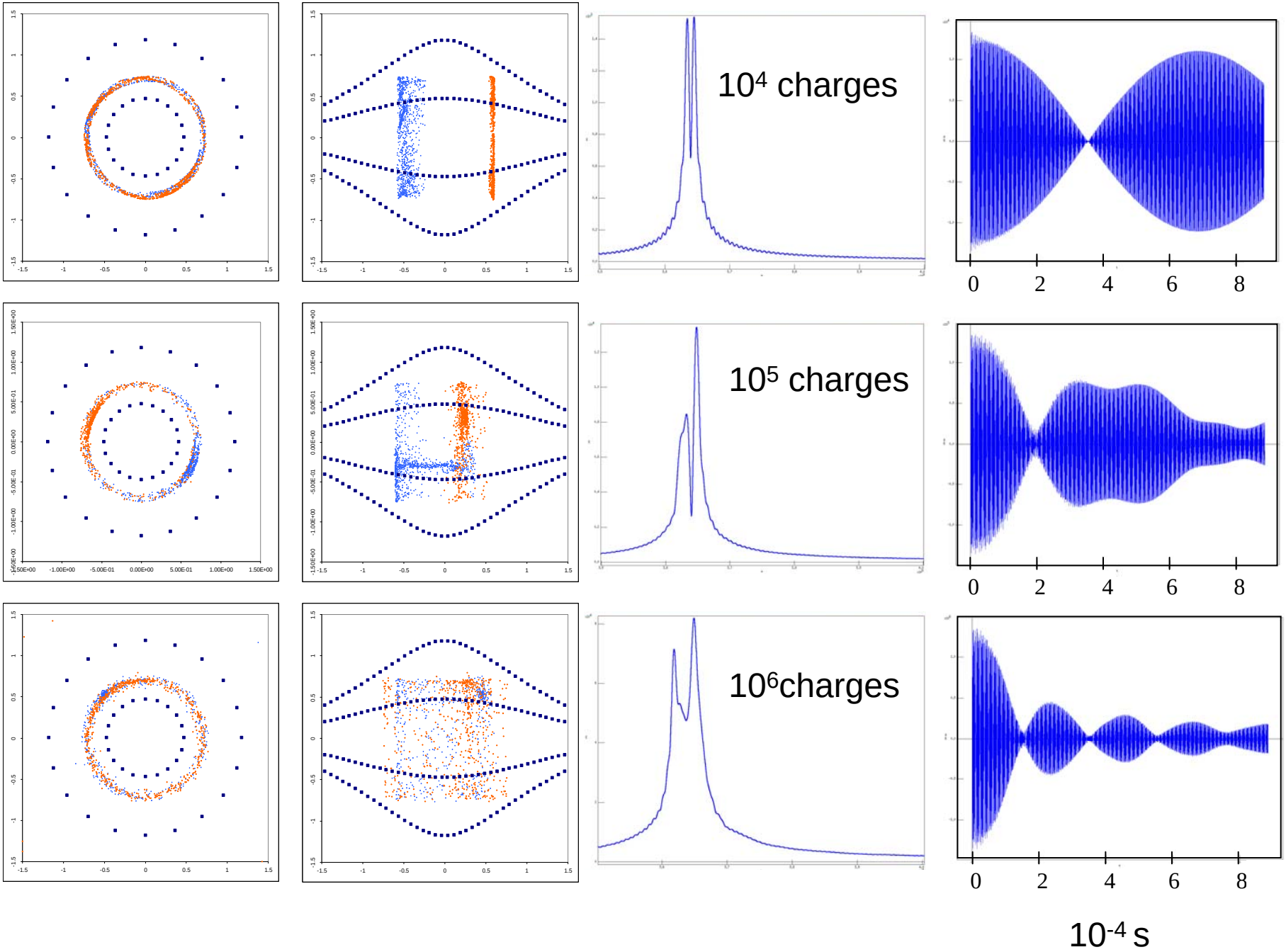


0.0 s

0.5 10^{-4} s

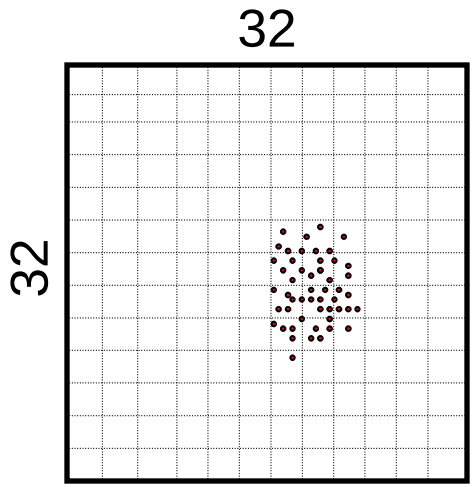
1.1 10^{-4} s

4.3 10^{-4} s

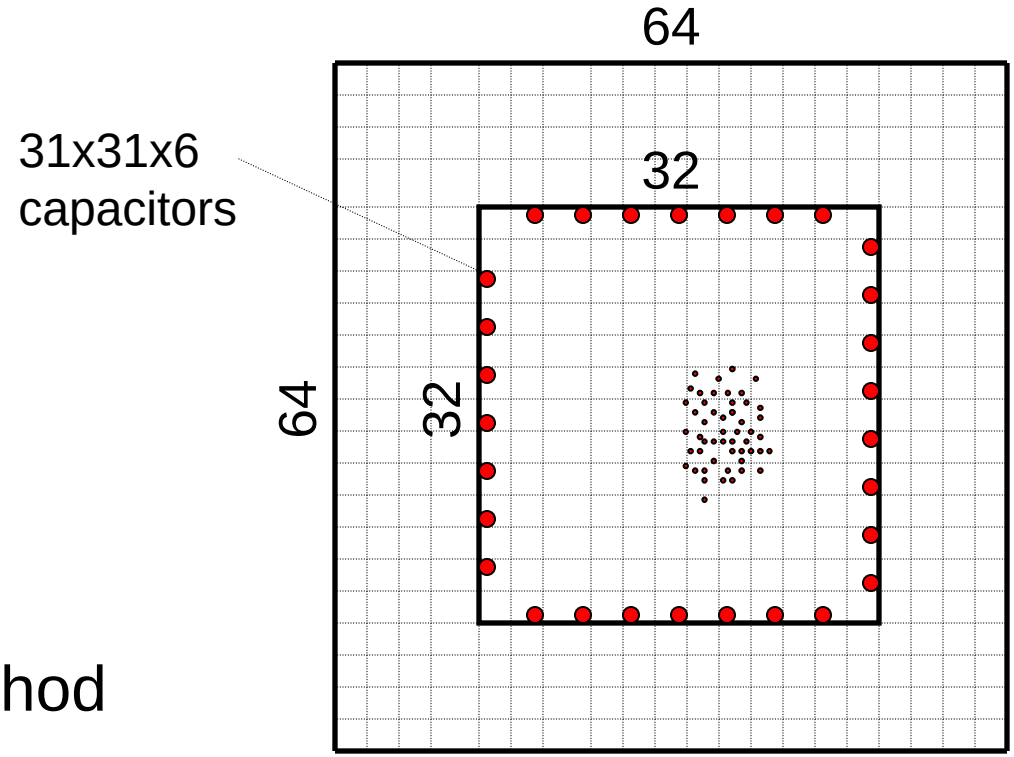

10^4 charges
10^5 charges
10^6charges
0 2 4 6 8
0 2 4 6 8
0 2 4 6 8
10^{-4} s

Comparison of different approaches

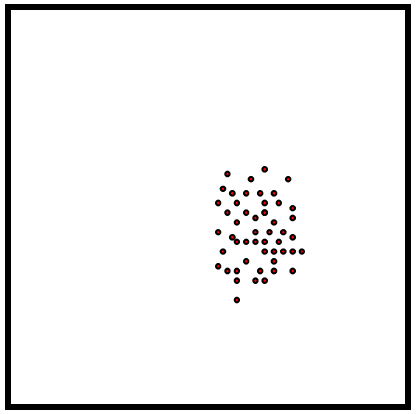
Particle in Cell Method



Capacity Matrix Method



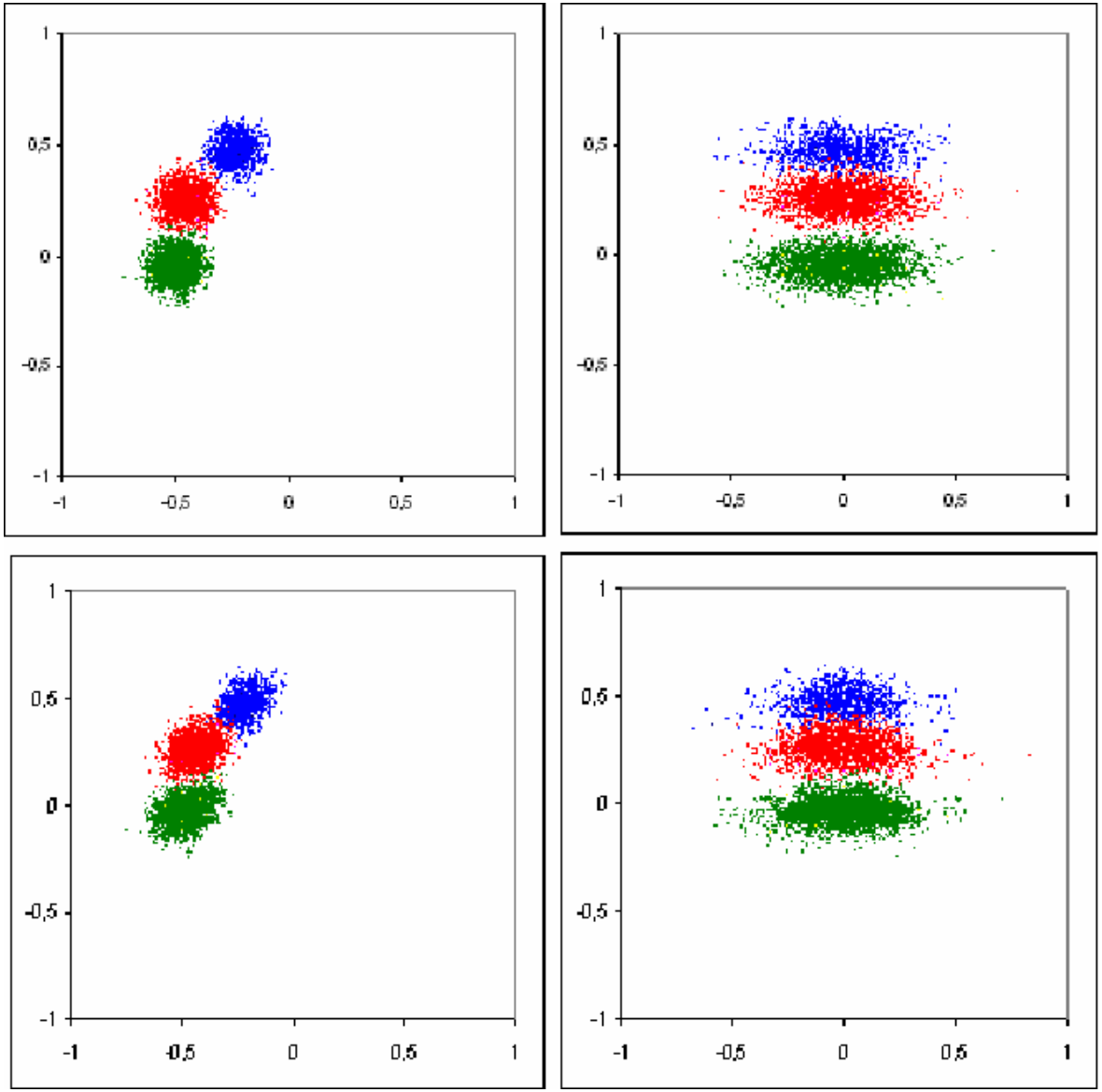
Particle-Particle Method



Particle in Cell versus Capacity Matrix Method

PIC

CM



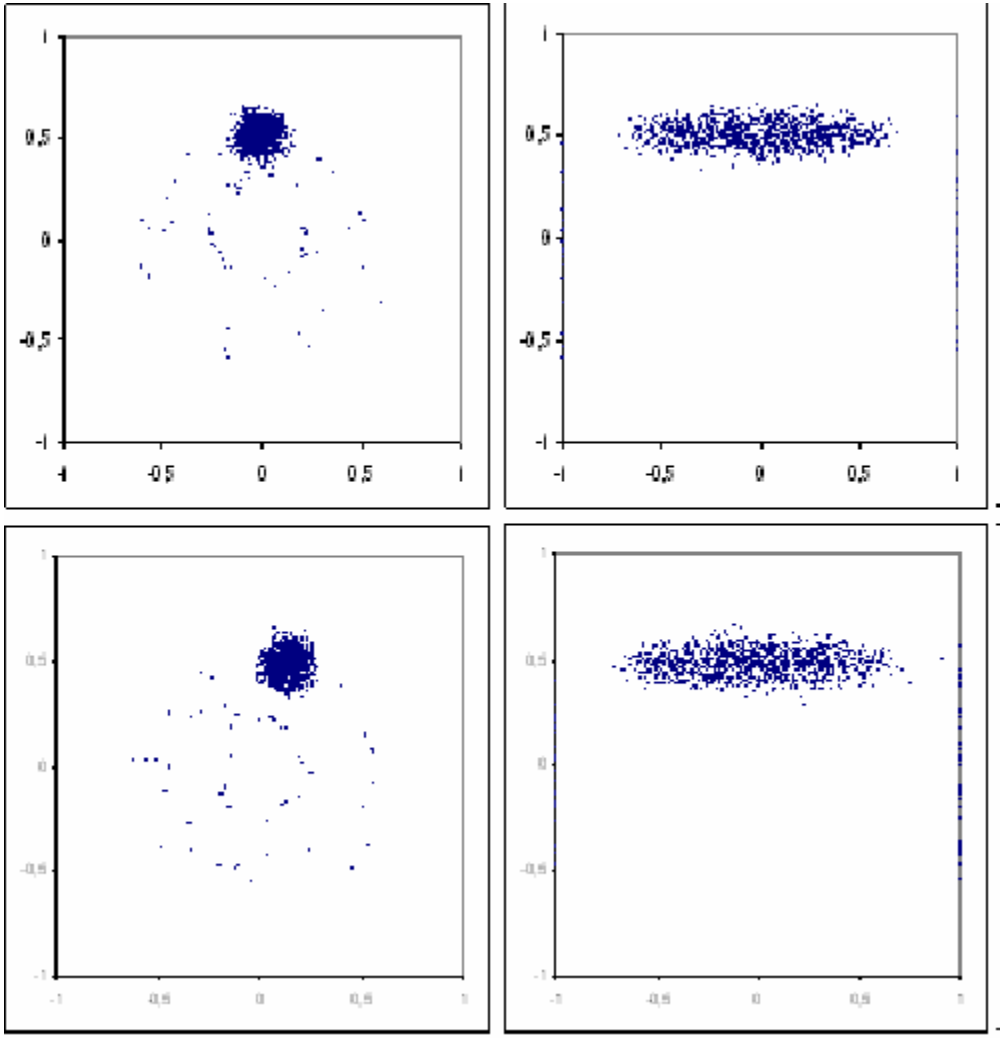
t = 210 us

1000 ions m = 99.7, z =1(blue), 2000 ions m = 100(red) z = 1,
3000 ions m = 100.3, z = (green) in a 1T magnetic field

Particle in cell versus Particle-particle method
(Small excitation time)

PIC

Particle-particle



T = 885us

1000 ions

m = 100000 Da

z = **<u>1000</u>**

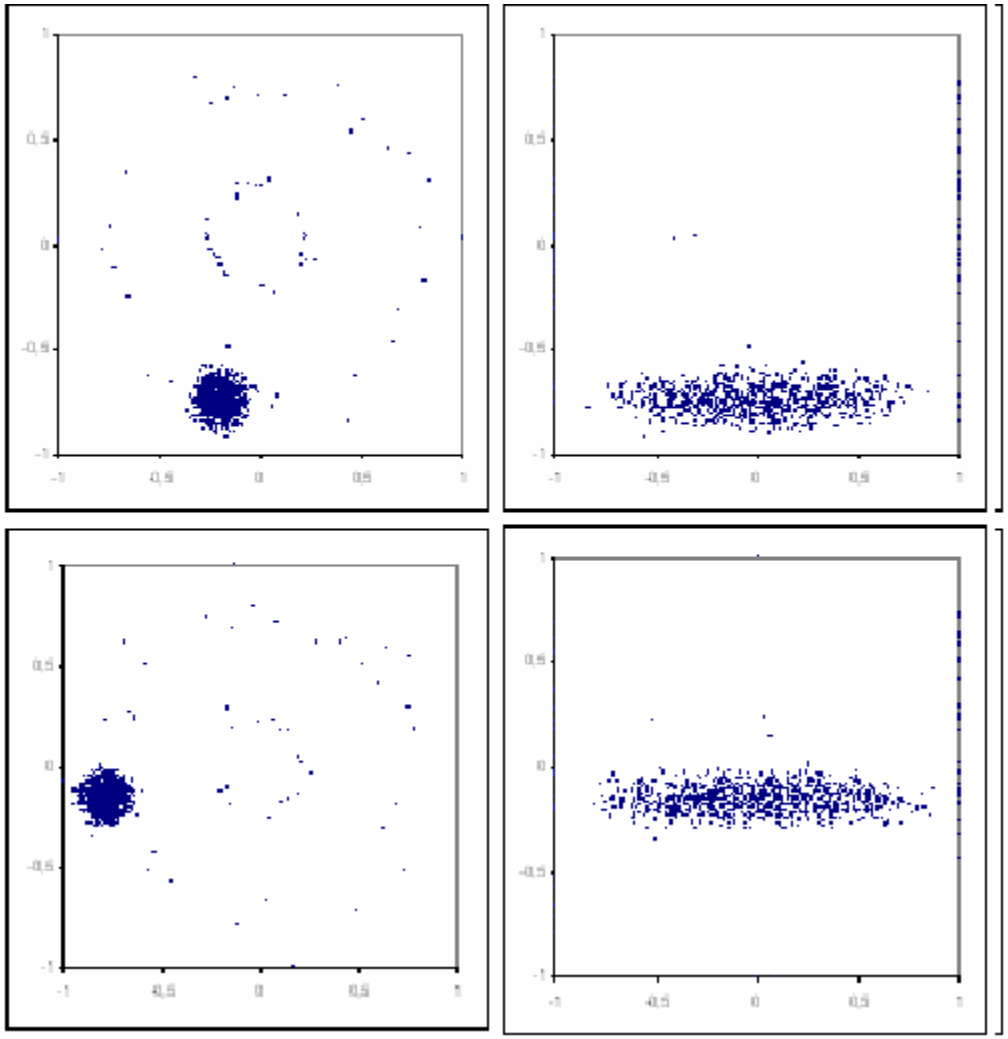
Excitation time **<u>40us</u>**

Particle in cell versus Particle-particle method (large excitation time)

Interaction with image charges is included

PIC

Particle-particle



T = 885us

1000 ions

m = 100000 Da

z = **<u>1000</u>**

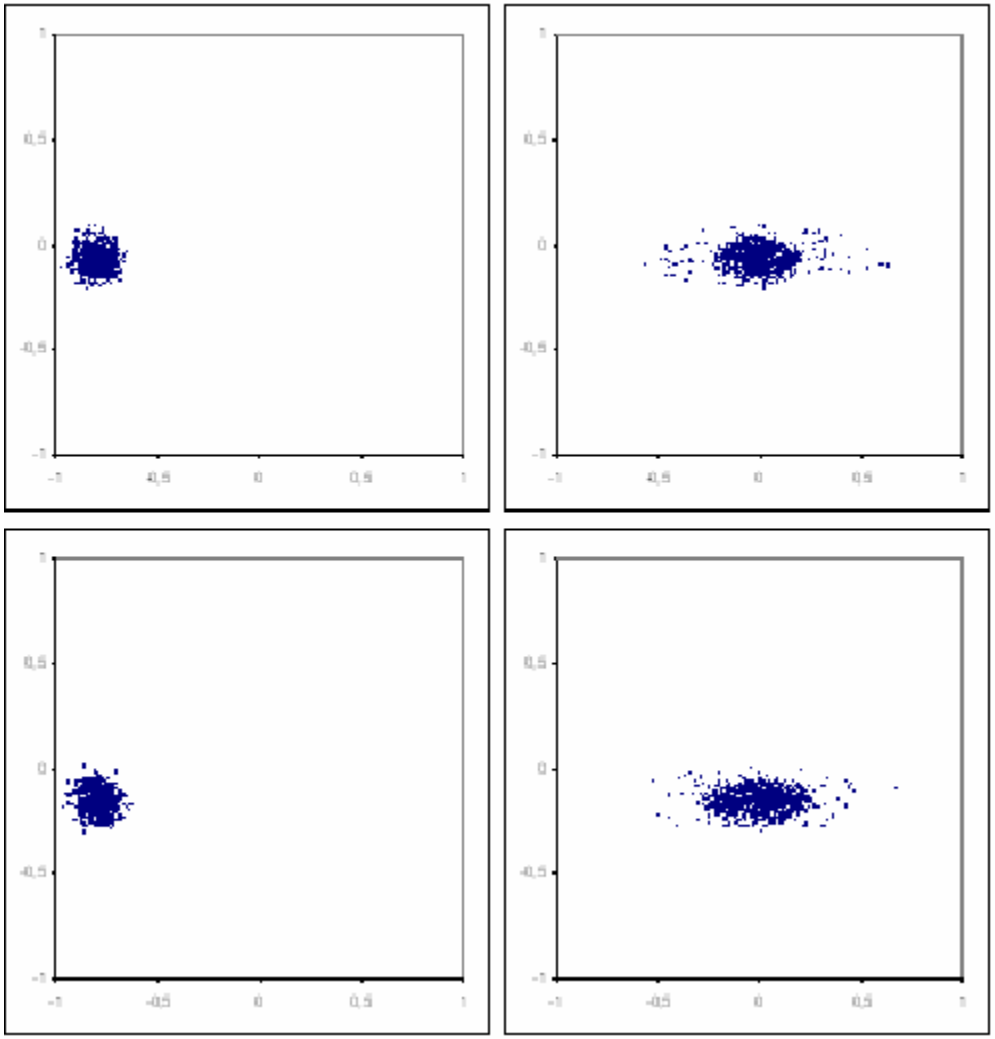
Excitation time **<u>70us</u>**

Interaction with image charges is not included

Particle in cell versus Particle-particle method
(large excitation time, small number of charges in the cell)

PIC

Particle-particle



T = 885us

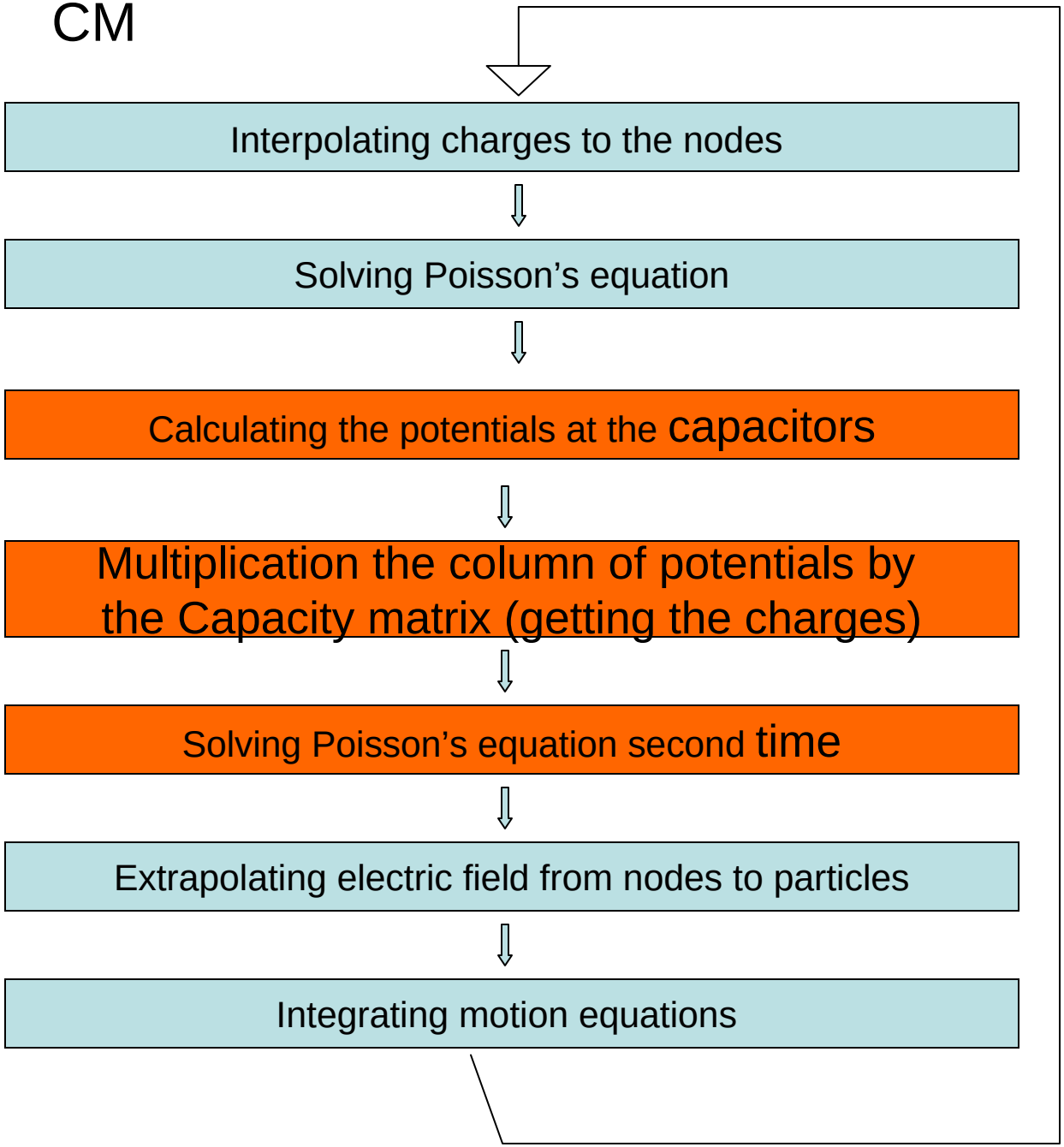
1000 ions

m = 10000 Da

z = **<u>100</u>**

Excitation time **<u>70us</u>**

CM


Interpolating charges to the nodes
Solving Poisson's equation
Calculating the potentials at the capacitors
Multiplication the column of potentials by the Capacity matrix (getting the charges)
Solving Poisson's equation second time
Extrapolating electric field from nodes to particles
Integrating motion equations

1 second transient corresponds to 15 million time steps.

Method	Number of capacitors	Time required for 1s transient
PIC		310 hours
CM	294	1151 hours
CM	726	1373 hours
CM	1536	1240 hours
CM	3174	2569 hours
CM	5766	5183 hours

The bottleneck procedure is multiplication by capacity matrix. It costs 0,9 s with the whole cycle 1,17 s in case of 5766 capacitors

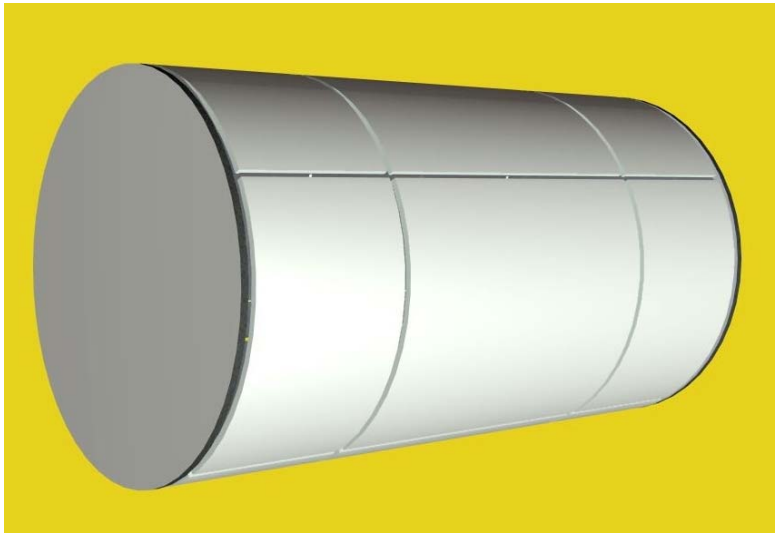
Capacity matrix method is extremely time consuming!

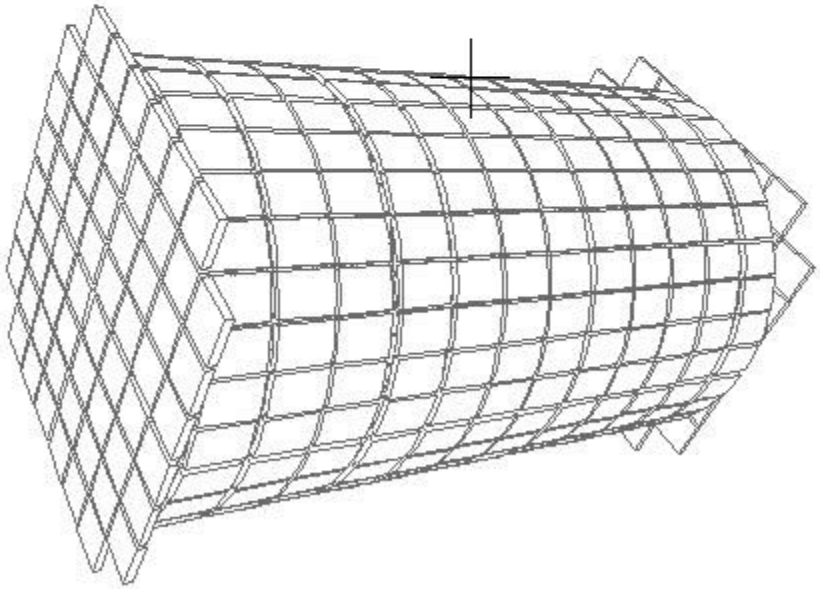
The new approach is to use it only for image charge field calculation

The field from electrodes could be calculated in advance by other methods

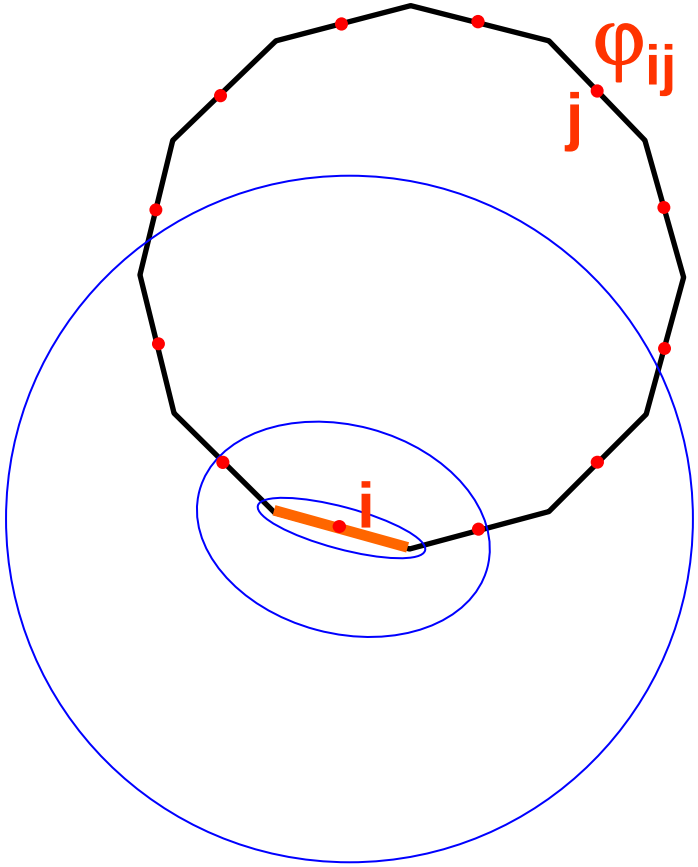
Surface Charge Method is the most accurate from known field calculating methods

The surface of the electrodes is divided into a number of square (or triangle) - shaped elements



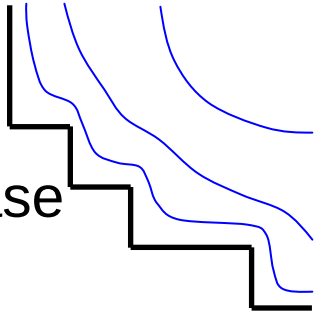


Surface Charge Method

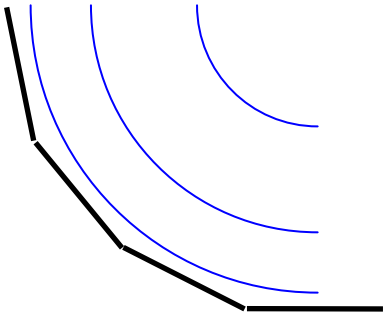


The potentials, each element (charged with unit charge) creates in the central points of all the other elements φ_{ij} are stored in memory. Then the system of linear equations is solved to find the charges of the elements needed to set the central points to desired potential. The charge is supposed to be distributed uniformly on each element.

Simion



SCM



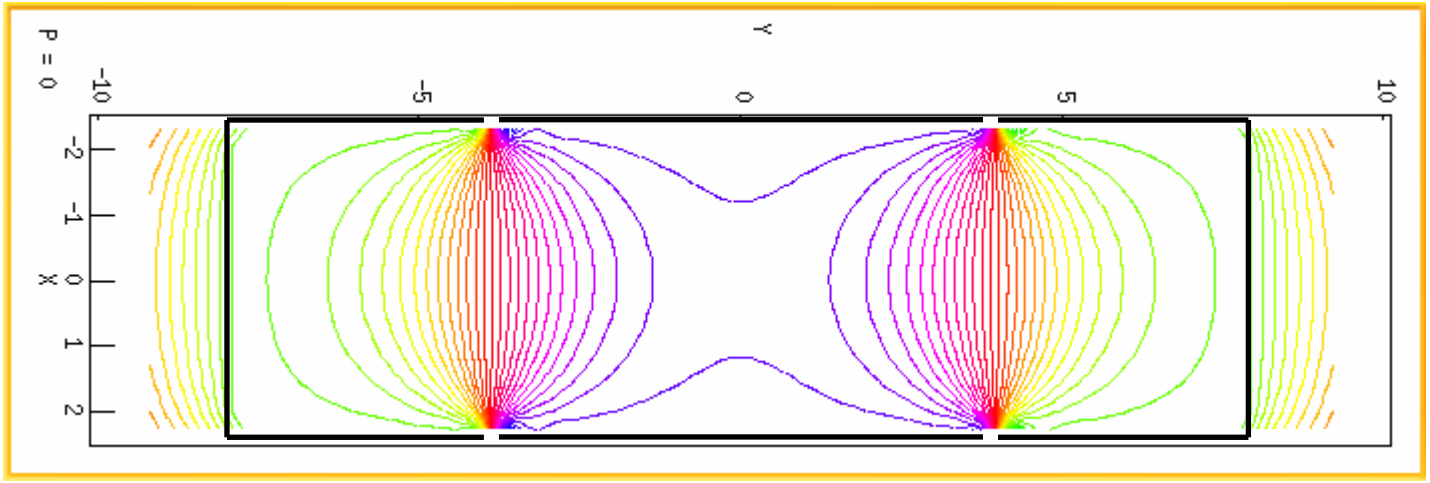
The surface of the electrode is simulated more accurately then in case of mesh based methods (Simion).

The algorithm

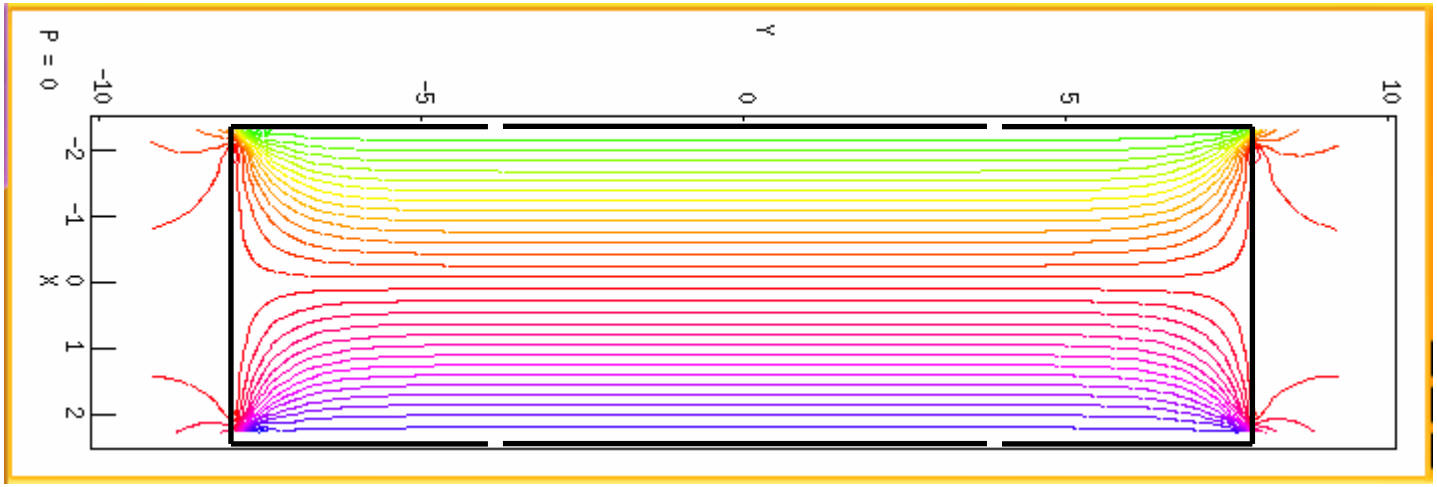
Our algorithm consists of two parts:

- Calculate the electric field created by the electrodes precisely, using the Surface Charge Method (done once before the main loop).
- Simulate ion motion inside this field, while electric field created by the ions is calculated using Particle-In-Cell approach combined with the capacity matrix method (at each time step).

Electric fields found by the use of SCM



ϕ_1

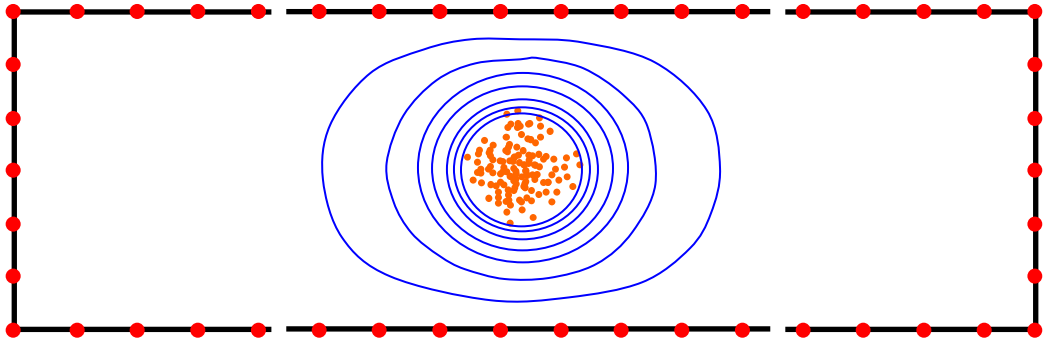


ϕ_2

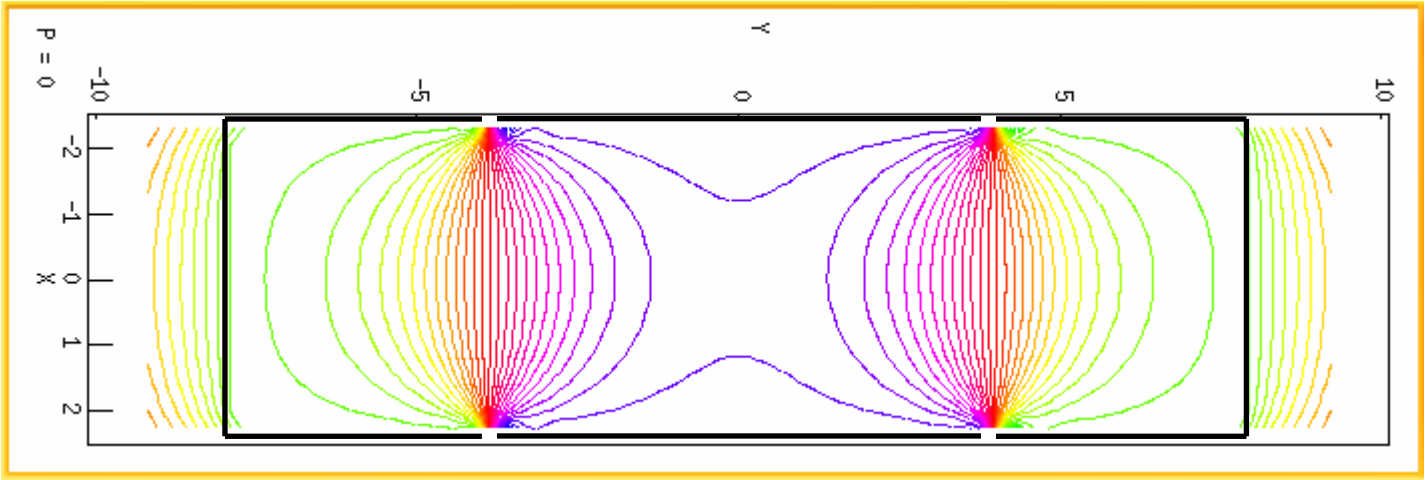
Excitation: $\phi = \phi_1 + \phi_2 \sin(\omega t)$

Detection: $\phi = \phi_1$

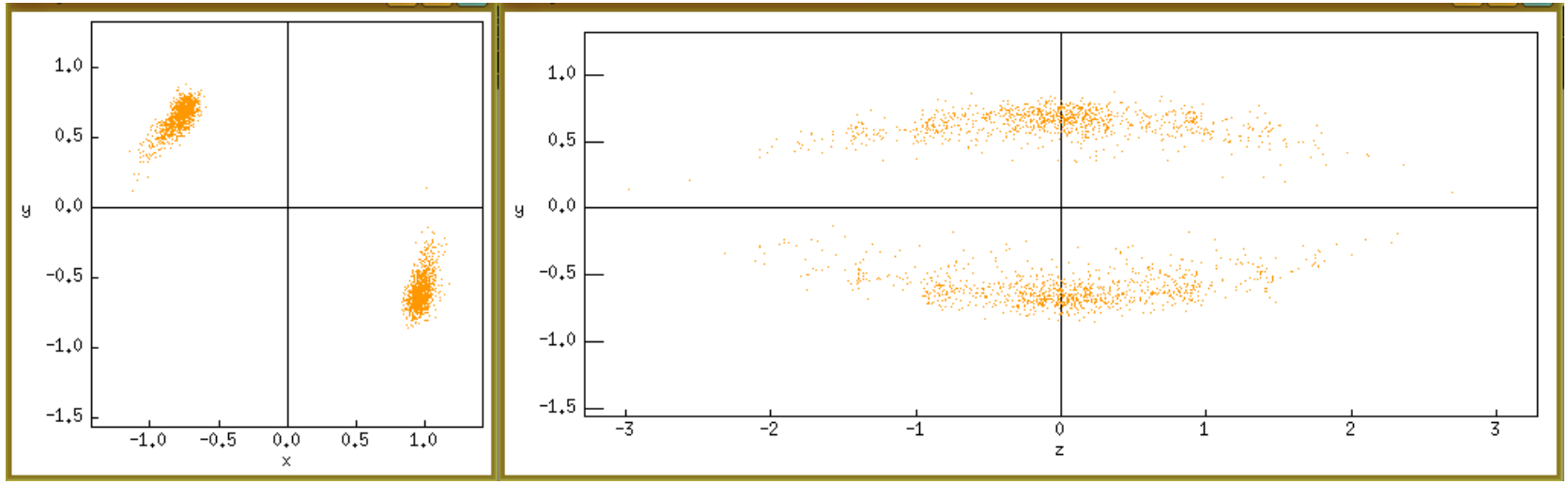
Field of the ions found by the capacity matrix method



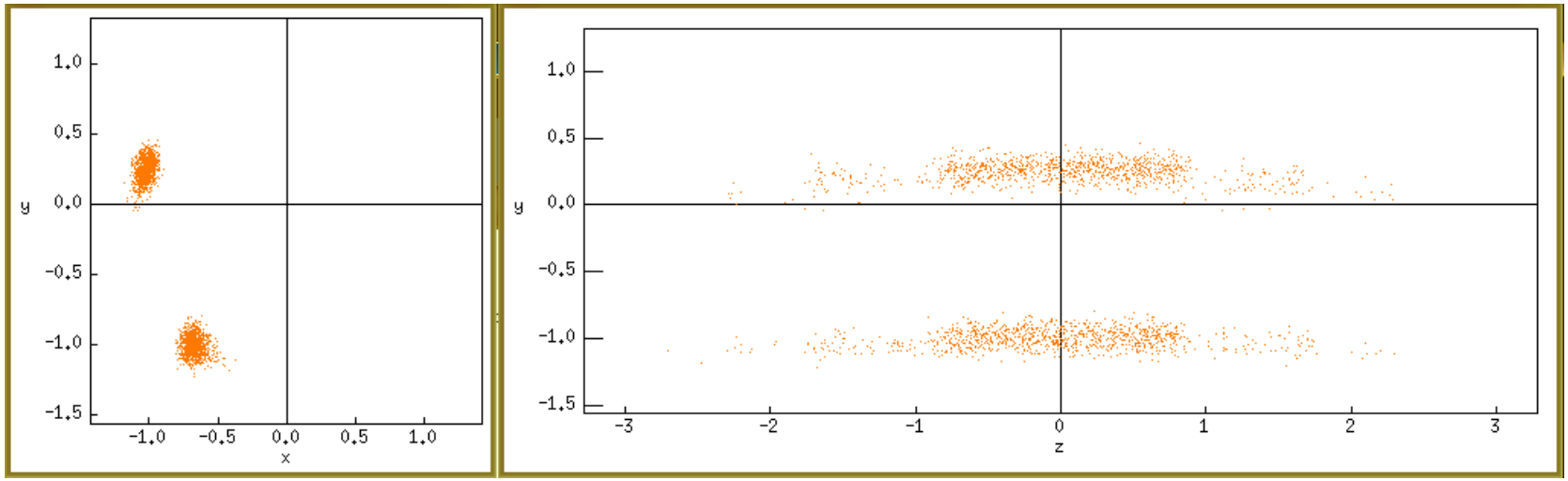
is added to the field of the electrodes found by SCM



Two ion clouds: 100 Da and 100.5 Da.



1.5 ms



3 ms

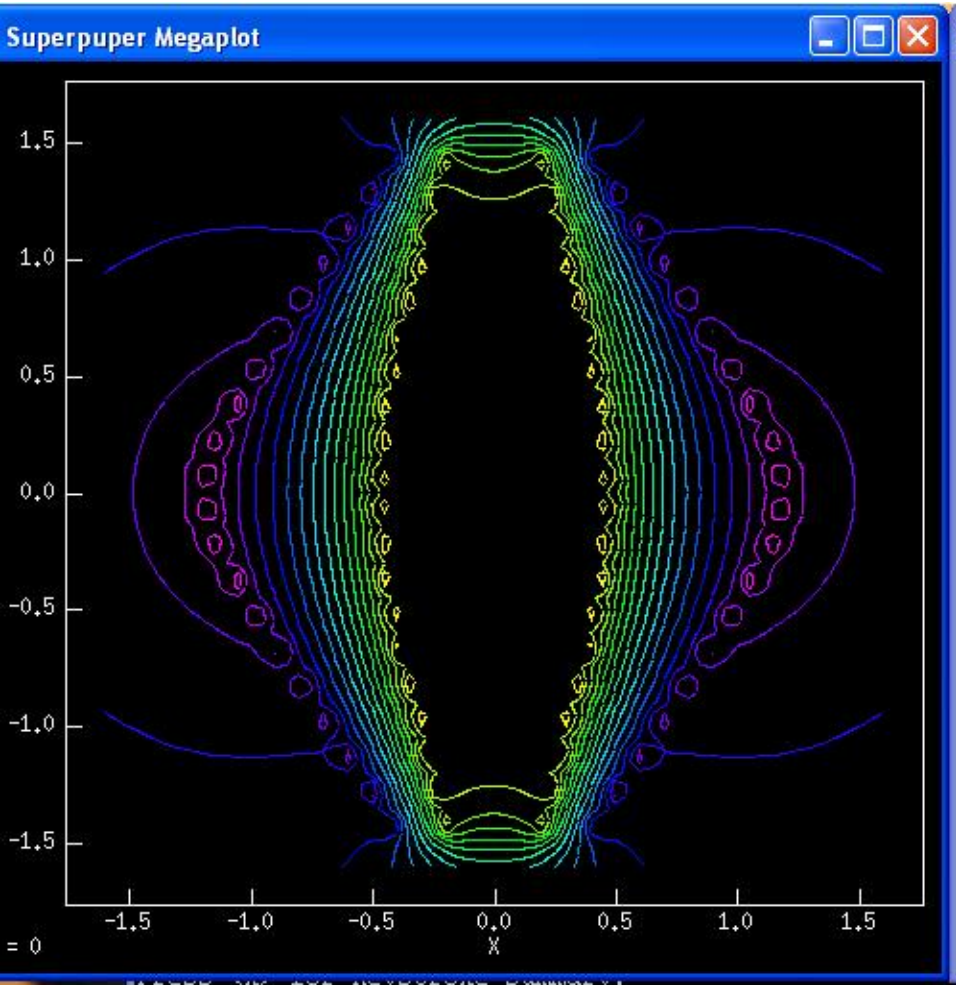
Computational time

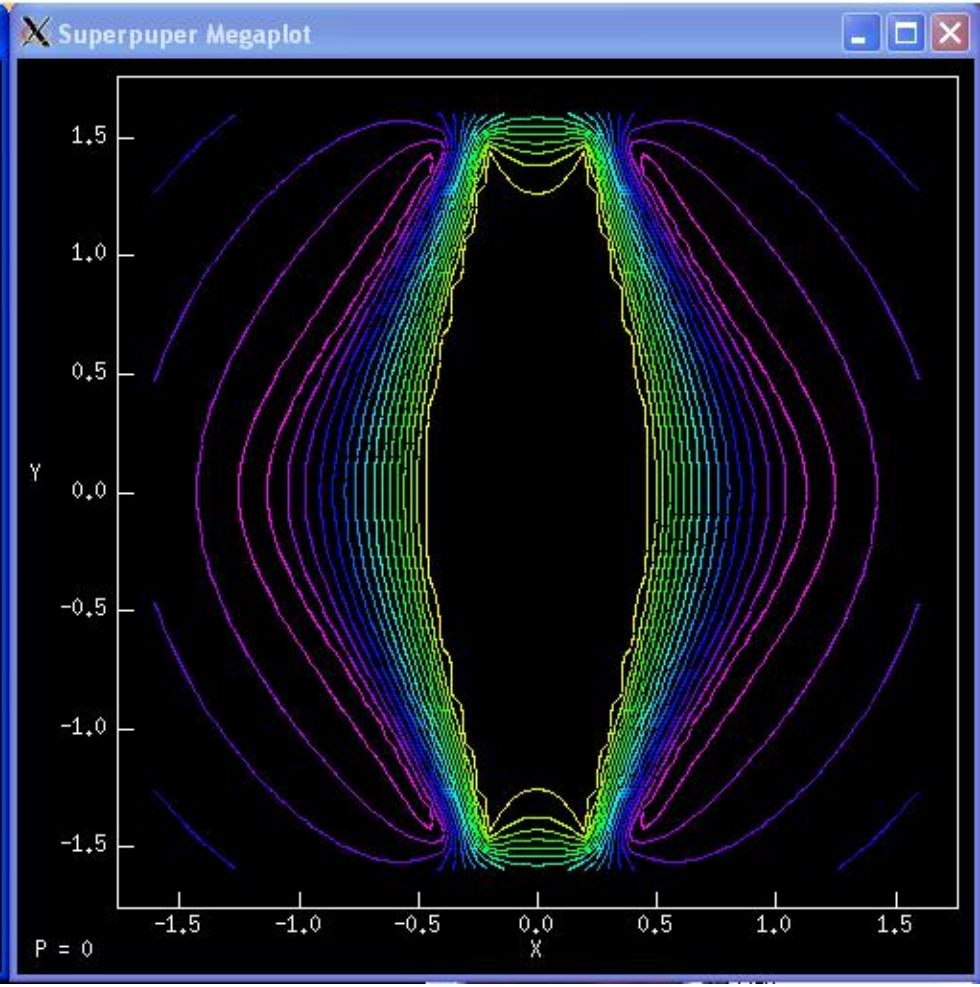
Number of nodes (capacitors)	1728
The grid for FFT Poisson solver	32x32x128
The number of ions	2000

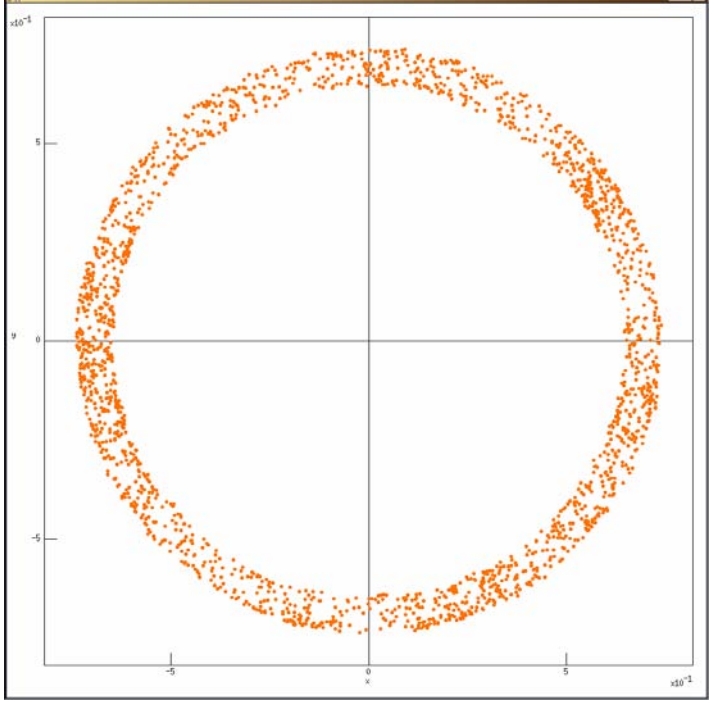
Most time-consuming procedures and total computational time (one processor 1.3 GHz)

Solving Poisson equation by FFT (executed twice each step)	0.017 s
Multiplication of the capacity matrix by the vector of potentials	0.045 s
Total time for one step	0.086 s

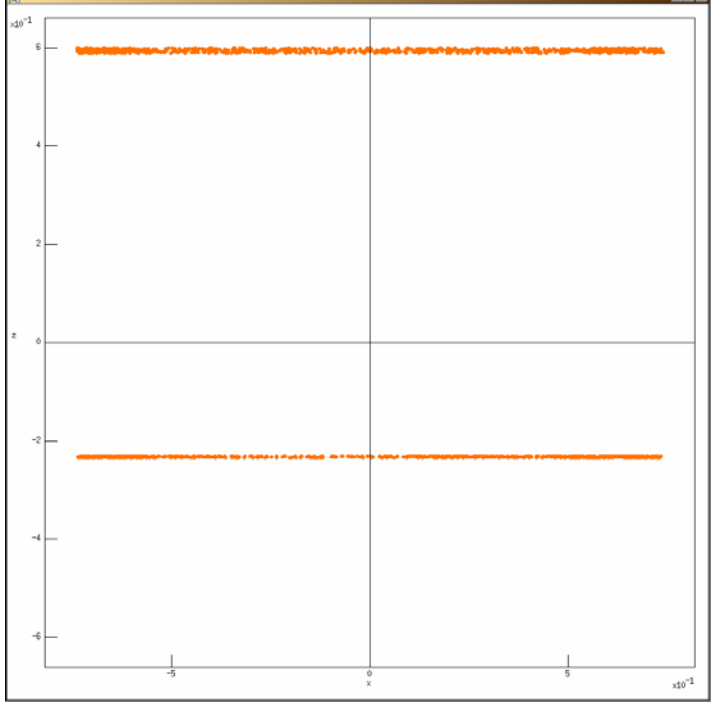
Kingdon trap fields calculated by capacity (left) and surface charge (right) methods

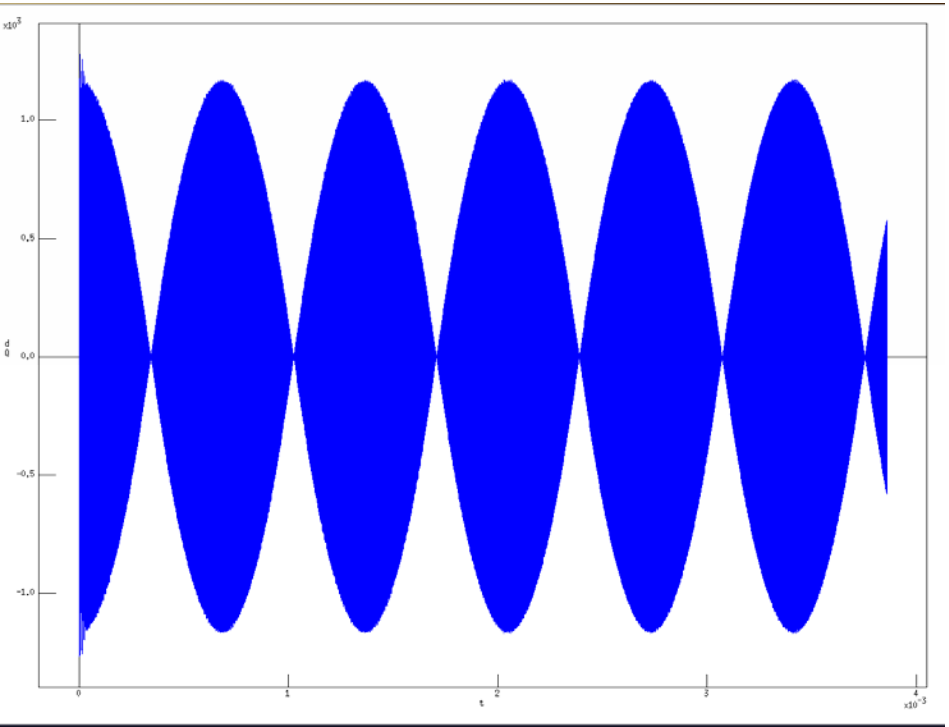


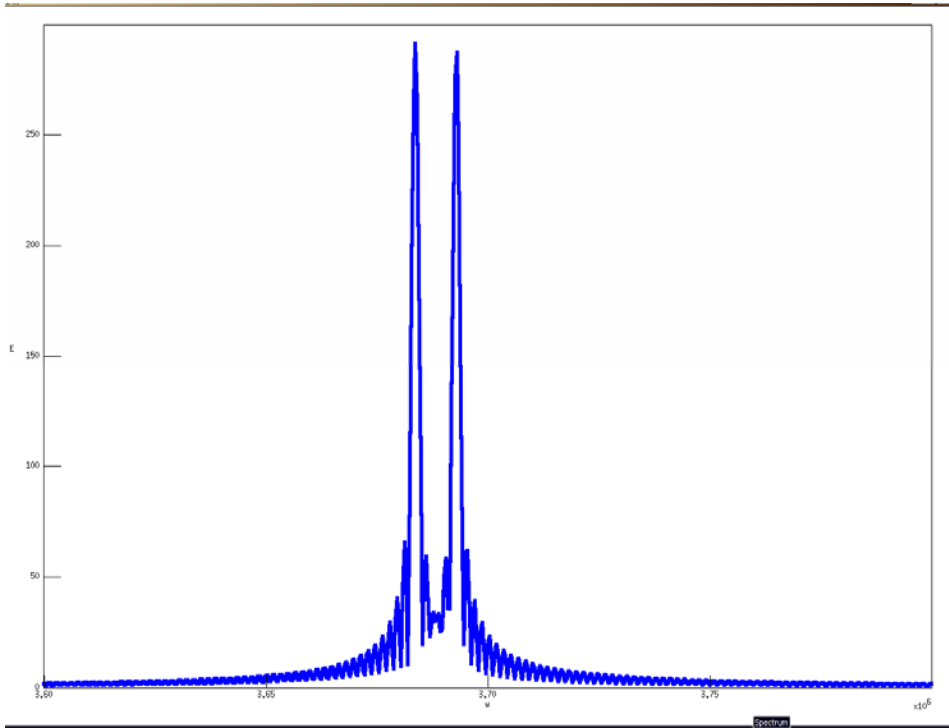


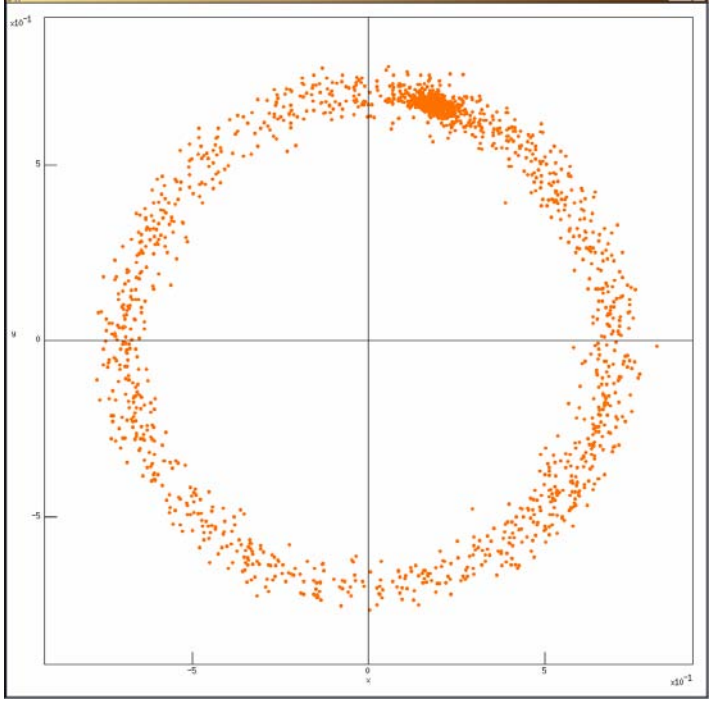


10^3 ions

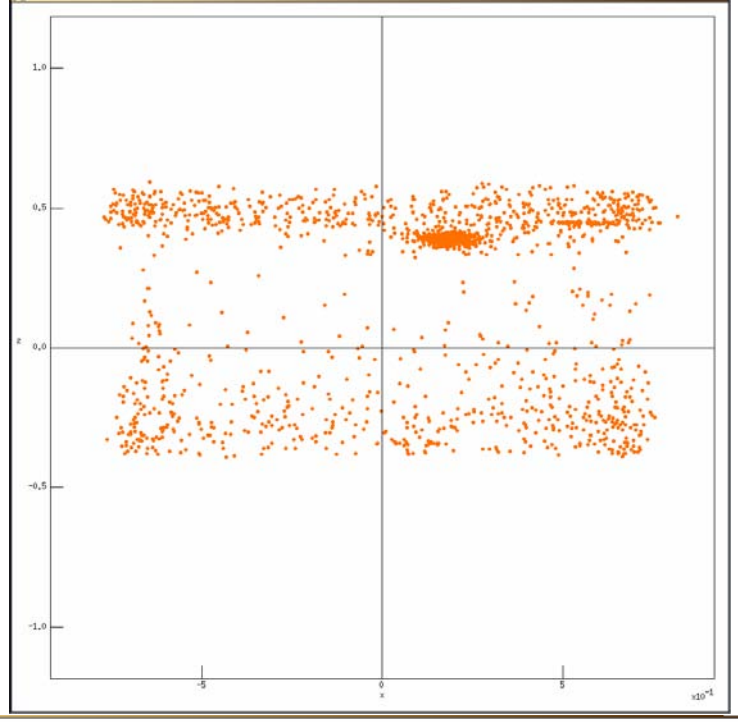


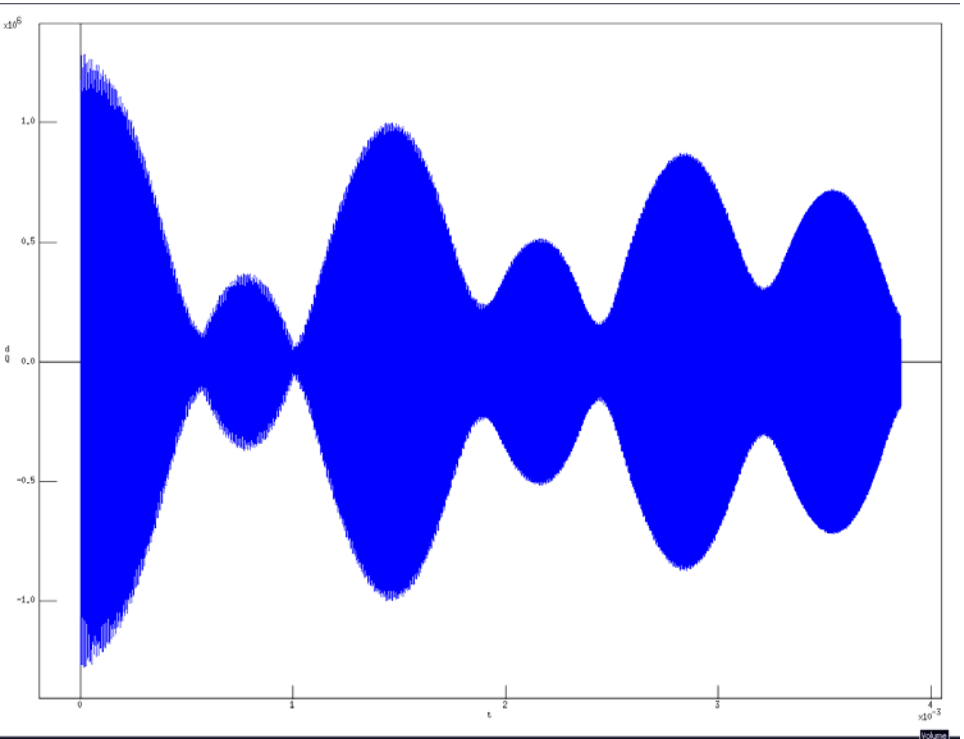


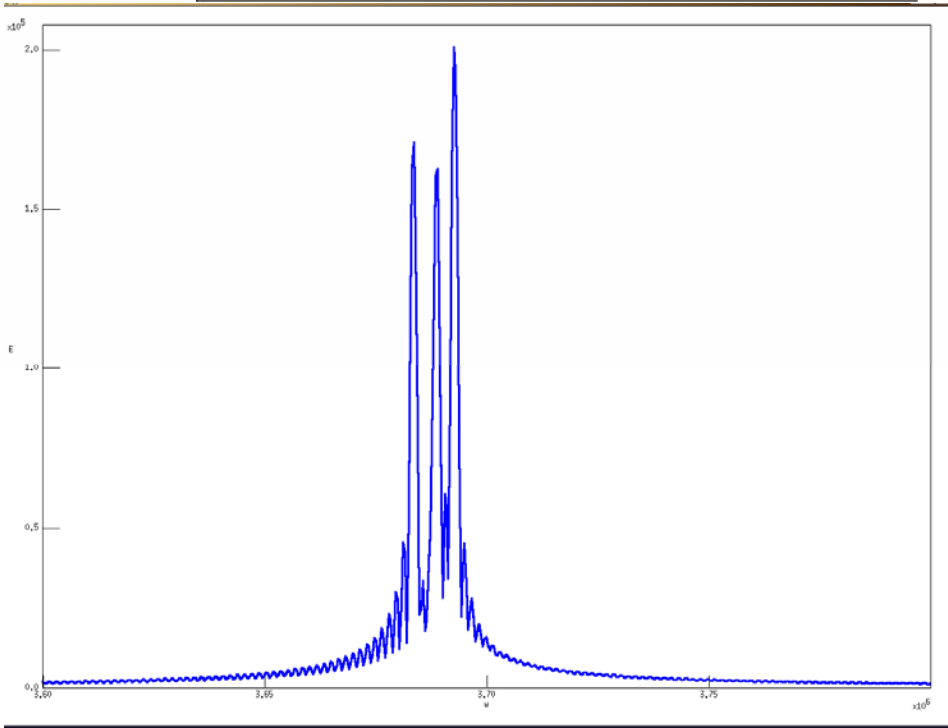




10^6 ions







Conclusions

- The new tools to simulate of ion cloud dynamics based on Particle in cell, particle-particle, capacity matrix and surface charge methods were developed for different electrode geometries and implemented on a supercomputer cluster
- Trajectory simulation for the ion clouds containing up to millions ions is possible in “reasonable” time scale
- Pronounced “charged plasma” effects were demonstrated in different ion traps

Future work

Code optimization and application to practical problems

Acknowledgements

Ron M.A. Heeren,

Alexander M. Popov,

Alexander V. Pozdneev,

Konstantin S.Chingin,

Pavel Ruymin

Dmitrij Avtonomov

Gleb Vladimirov

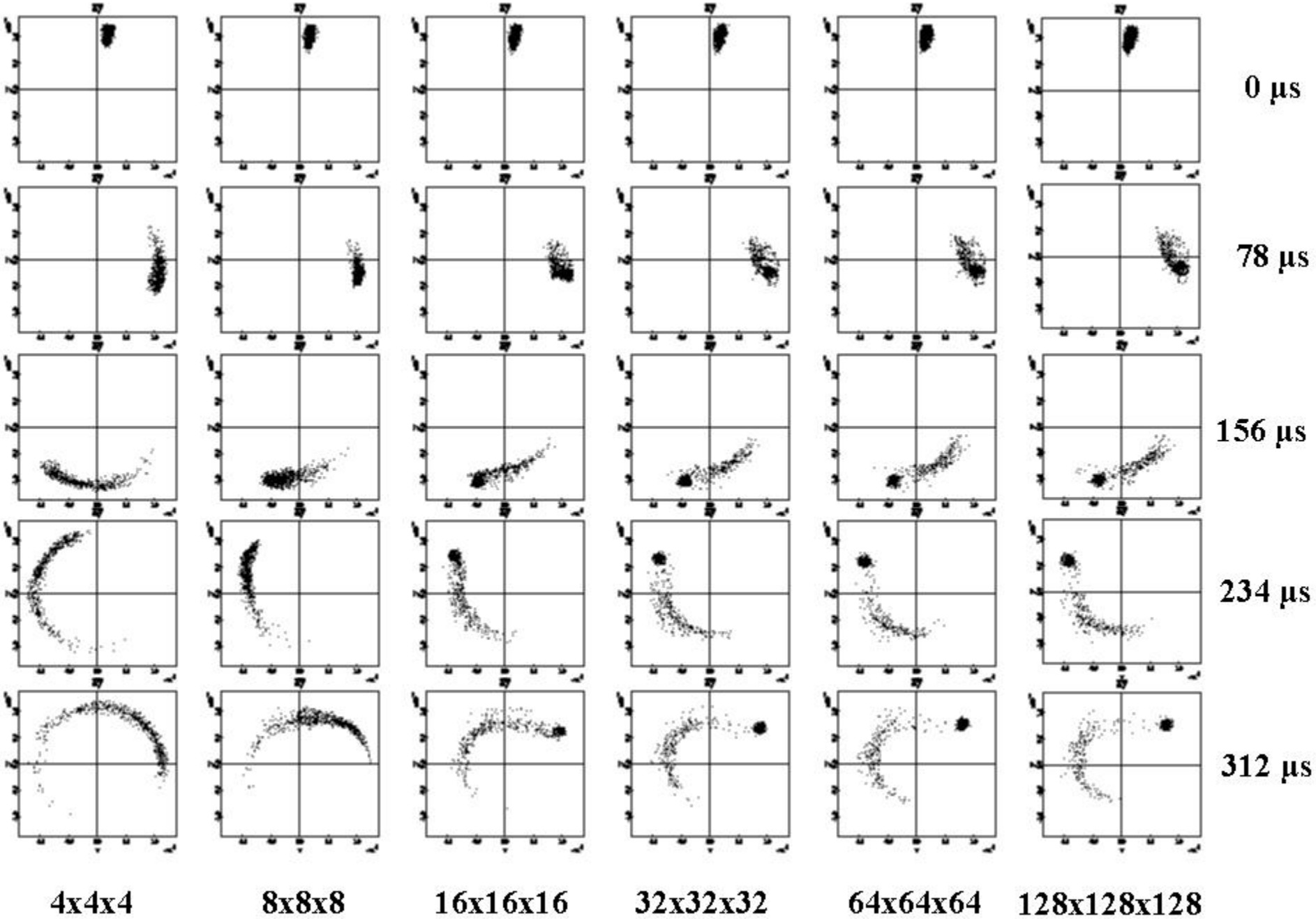
Ivan Boldin

NWO-RFBR 04-02-89000-HBO_a

Architecture

Every node in Aster is a CC-NUMA machine. The term CC-NUMA stands for Cache-Coherent Non Uniform Memory Access. In the CC-NUMA model, the system runs one operating system and shows only a single memory image to the user eventhough the memory is physically distributed over the processors. Since processors can access their 'own' memory (i.e. memory on the same physical board as the processor) much faster than that of other processors, memory access is non uniform (NUMA). In this architecture the contents of the various processor caches should be coherent requiring extra hardware and a cache coherency protocol. A NUMA computer fulfilling these requirements is called a CC-NUMA machine.

Consistency test


0 µs
78 µs
156 µs
234 µs
312 µs
4x4x4
8x8x8
16x16x16
32x32x32
64x64x64
128x128x128