

# Analyzing Complex FTMS Simulations: a case study in high-level visualization of ion motions

W. Burakiewicz\*

Center Mathematics and Computer Science  
Amsterdam

R. van Liere †

Center Mathematics and Computer Science  
Amsterdam

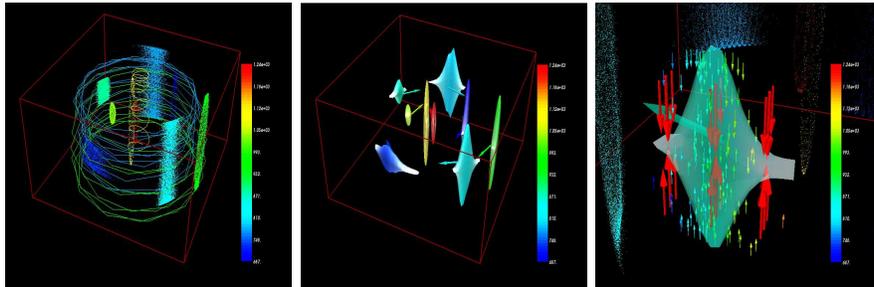


Figure 1: Three views of ion-motion: Left is a low-level direct rendering of an ion-cloud. Trajectories are used to show motion of a few ions. Middle image shows 'comet icons' which encode motion and density properties of clusters of related ions. The right image shows relative ion motions from a camera tethered to the center of the comet.

## ABSTRACT

Current practice in particle visualization renders particle position data directly onto the screen as points or glyphs. Using a camera placed at a fixed position, particle motions can be visualized by rendering trajectories or by animations. Applying such direct techniques to large time dependent particle data sets often results in cluttered images in which the dynamic properties of the underlying system are difficult to interpret.

In this case study we take an alternative approach for the visualization of ion motions. Instead of rendering ion position data directly, we first extract meaningful motion information from the ion position data and then map this information onto geometric primitives. Our goal is to produce high-level visualizations that reflect the physicists' way of thinking about ion dynamics. Parameterized geometric icons are defined to encode motion information of clusters of related ions. In addition, a parameterized camera control mechanism is used to analyze relative instead of only absolute ion motions.

We apply the techniques to simulations of Fourier transform mass spectrometry (FTMS) experiments. The data produced by such simulations can amount to  $5 \cdot 10^4$  ions and  $10^5$  time steps. This paper discusses the requirements, design and informal evaluation of the implemented system.

**CR Categories:** I.3.6 [Computer Graphics]: Interaction Techniques; I.6.6 [Simulation and Modelling]: Simulation Output Analysis

**Keywords:** particle visualization, motion, motion features

\*e-mail: burakiew@cwi.nl

†e-mail: robertl@cwi.nl

## 1 INTRODUCTION

Complex physical models involving large number of particles are often investigated with the aid of computer simulations. Simulations allow the researchers to gain insight into the model, and help them design optimal experimental setups or complete instruments, such as mass spectrometers or particle accelerators. Particle simulations usually produce huge amounts of data. Due to the inherent complexity of the particle motions, it is often necessary to use visualization techniques for its analysis to interpret the simulation results. Visualization would broaden the understanding of the underlying model and help explain the resulting phenomena.

Many visualization systems have been developed to cope with particle simulation data. The most straightforward visualization technique is to render particle position data directly as a point in three-dimensional space. Particle properties such as mass, velocity, charge or energy may also be incorporated as attributes of the point. Particle motion can be visualized by drawing trajectories for each particle or by animation. Other techniques, such as particle subset selection or rendering glyph representations of particles, can be employed to simplify the image. Finally, the user can explore the resulting point cloud in a perspective projection using basic camera movements: rotation, panning and zooming.

A major problem with the direct rendering of large particle data sets is that the resulting images depict only low-level particle position data and contain no explicit semantical information of the particle motions. The resulting animations can be very cluttered and it is left to the user how to interpret the simulated phenomena from the images. The goal of this case study is to produce animations that reflect a mass spectrometry physicists' way of thinking about ion dynamics. By understanding what motion information physicists want to see from an ion data set, visualization techniques that depict more high-level semantical concepts can be developed.

In this case study, we take an alternative approach for the visualization of ion motions. Instead of rendering ion position data directly, we first extract meaningful motion information from the ion position data and then map this information onto geometric primitives. We have applied our particle visualization system for the

analysis of Fourier transform mass spectrometry (FTMS) simulations. Data sets produced by such simulations can be large:  $5 \cdot 10^4$  ions and  $10^5$  time steps are typical data set sizes. We employ two basic techniques: parameterized geometric icons and parameterized camera control, in which parameters are derived by extracting motion properties from the data. Geometric icons are used to simplify the image while retaining the essential motion properties of a related cluster of ions. Camera control can be used to analyze relative (instead of absolute) ion motions and to visualize very specific motions that are of interest to the mass spectrometry researcher. For example, consider Figure 1. The left panel shows an ion data set as a direct rendering; some ions are shown with their trajectories. In the middle panel, ion clusters are displayed as geometric icons. The shapes of the icons resemble comets illustrating the varying ion density. The right panel shows a parameterized camera view; the camera is positioned relative to the comet allowing for relative motions of ions within a cluster to be viewed.

### 1.1 Fourier transform mass spectrometry

Mass spectrometry is a powerful analytical technique that is used to identify unknown compounds, to quantify known compounds, and to elucidate the structure and chemical properties of molecules. The power of mass spectrometry is that compounds can be detected and identified at very low concentrations in chemically complex mixtures.

Fourier transform mass spectrometry (FTMS) is a particular technique used in mass spectrometry. For FTMS devices to operate robustly, very accurate detection of mass to charge ratio ( $m/z$ ) of ions trapped in an electromagnetic field is required. The more accurate the ions can be detected, the more resolving power can be achieved. The aim of FTMS simulations is to obtain insight into the dynamics of ion motions and to study how initial conditions can influence the ion behavior.

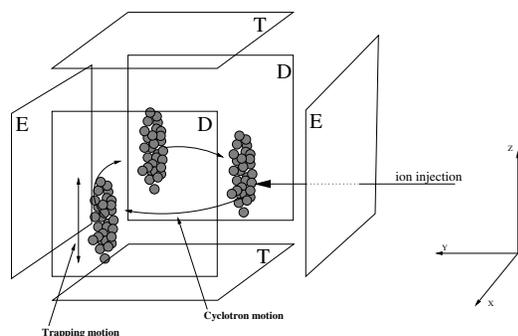


Figure 2: Schematic view of a Fourier transform mass spectrometer cell.

The central part of a FTMS model is a cubic vacuum cell. The sides of the cell are called trapping, excitation and detection plates (marked as T, E and D on Figure 2). Ions remain trapped in the cell: static magnetic field parallel to the Z-axis prevents the ions from escaping in X and Y directions. By maintaining electric potential between the trapping plates, the ion movements are also constrained in the Z direction. There are three phases of the experiment. During cooling phase, ions are injected into the cell and cooled to room temperature. Then, in excitation phase, an electric field perpendicular to the magnetic field is generated between the excitation plates. This causes the ions with the same  $m/z$  ratios to form clusters and oscillate with a frequency depending on their  $m/z$  value. During the detection phase, the oscillating ion clusters induce a signal on the detection plates.

If the initial conditions of the experiment are optimally set, then each  $m/z$  ion cluster will result in a distinguishable peak in the signal's Fourier spectrum. The peaks in the signal are used to discern the ion species (different  $m/z$  values) within the cell. In practice however, different peaks tend to merge and the ion species cannot be distinguished. During the experiment it is important that the ions move in phase coherent clusters. If ion clusters are not coherent (i.e. dephase) the signal cannot be detected as a single peak. The initial conditions that can influence the detection accuracy are the excitation frequency profile and time, magnetic field strength, density of ions, and initial temperature.

We list a number of phenomena that are important for the mass spectrometry physicist to obtain insight into the simulation output:

1. **Specific ion motions:** ion motions can be split into three separate motions that emerge from well known physical laws [6]:
  - *cyclotron motion* – the circular motion in the XY plane. In theory, the cyclotron motion frequency of an ion is expressed as:  $\omega_c = \frac{zB}{m}$  in which  $B$  is the static magnetic field,  $m$  is the ion mass and  $z$  is the ion charge. Cyclotron frequencies in our studies are in 100 - 200 kHz range.
  - *trapping motion* – oscillation along the Z axis due to the trapping potential between the trapping plates (see Figure 2). The trapping motion frequency of an ion is expressed as:  $\omega_t = \sqrt{\frac{kz}{m}}$  in which  $k$  is a constant related to the quadratic z-potential. Frequency of this motion is 1 - 10kHz.
  - *magnetron motion* – also called  $B \times E$  drift. It arises as a second solution of the ion motion equation in the static magnetic and electric fields. The magnetron motion frequency is: 1 - 100Hz.
2. **Frequency perturbations:** Many factors can influence ion motions during the experiment and simulation. For example, ion cluster interactions can occur, ion-ion interactions within a cluster, non-optimal initial conditions, detection noise, etc, can influence the motion of (a cluster of) ions. Hence, the detected ion frequencies need not to be the same as the theoretical oscillation frequencies; i.e.  $\omega_c^{theory} - \omega_c^{measured} \neq 0$  for cyclotron frequencies. The question is: what causes the frequency perturbations?
3. **Cluster dephasing:** during the excitation phase, ions form phase coherent clusters. Over time, the clusters tend to dephase; i.e. phase differences between ions in a cluster tend to increase. Dephasing can be seen by studying the structural shape of the cluster. The structure of the ions in the cluster will gradually change from a coherent to a less coherent shape. The questions we want to answer are: Under what circumstances does dephasing happen and is it an instantaneous or progressive event?
4. **Phase locking:** two ion clusters with different  $m/z$  ratios should induce signals with different frequencies, even if their  $m/z$  are very close. However, clusters with close  $m/z$  ratios may start oscillating together with the same frequency and phase, and are therefore detected as a single cluster. This undesirable effect is called phase locking. What triggers the phase lock event?

## 2 RELATED WORK

Most of the existing particle visualizations directly render particle data as a point cloud. Traditionally, points, polygons, or icons are

used to represent particle positions. Particles are rendered as a point cloud in 3D, or in 2D scatter plots by projecting data onto the XY, XZ and YZ planes. For example, Figure 3 shows a 2D-scatterplot of a time step from the FTMS simulation data. Particle properties such as mass, velocity, charge or energy are incorporated as attributes of the point. Particle motion can be visualized by drawing trajectories for each particle or by animation.

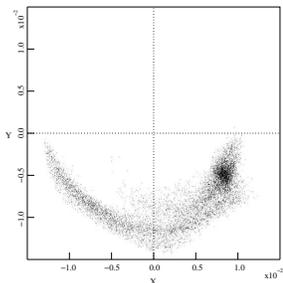


Figure 3: Direct rendering of ion cloud onto XY plane. Direct rendering of an ion cloud onto XY plane. Ions form a comet-like structure.

Due to space restrictions, we discuss two alternative approaches for representing particles:

- Co et al. visualized high-dimensional particle simulation data using a number of coupled, two and three-dimensional scatterplot views, [2]. As a mean of exploring physical data, disk glyphs and interactive selection, reflected in multiple views, were proposed. Brushing techniques are used for particle marking and the selection is immediately reflected in multiple views.
- Volume rendering was used to render particle trajectories, resulting from relativistic heavy ion collider (RHIC) experiments, [9]. The emphasis is put on displaying huge number of elementary particle trajectories created during ion collisions. The amount of data is reduced by different means of track filtering, range of interest specification and trajectory clustering. Nonlinear magnification (fish-eye camera) was proposed as a mean of displaying part of the data in detail while keeping the global view.

Both systems render particle position data directly as a point in three-dimensional space. The direct rendering results in only low-level particle position data and contain no explicit semantical information about the particle motions. It is left to the user on how to interpret the simulated phenomena from the images.

There have been many case studies that visualize high-level information of particle particle motions. For example, Huitema et. al. discuss techniques for the interactive 3D visualization of protein dynamics, [3]. They apply essential dynamics methods to filter out uninteresting atom motions from the larger concerted motions. In this way, clear and concise 3D animations of protein motions can be produced. Robbins et. al. have developed novel visual techniques to analyze the structural details of waves in response data from the turtle visual cortex, [8]. Space-time and latency maps were used to visualize structurally coherent motions. Finally, Parker et. al. have developed novel visualizations to study tokamak plasma turbulence, [7]. Poloidal and toroidal slices were used to display electrostatic potential over time. All these case studies use custom visualization techniques for depicting the physical phenomena of the underlying problem, however, it is not clear if and how these techniques can be generalized.

Besides novel visual representations of particle motion, much research is focused on dealing with the amount of data produced

by particle simulations. Much effort is put into designing a parallel rendering system to interactively display giga- to terabytes of data, [10, 5, 4]. The resulting systems are very robust in the terms of rendering speed and data processing. Our ion visualization system does not put much focus on large data sets. As we show in Section 4, our visualization system does not compute motion properties on the complete data set, but needs only a small history of time steps. We show that for medium-sized datasets (tens of GB) this can be achieved on a modern desktop computer.

### 3 VISUALIZATION IDIOMS

Since the cyclotron motion frequency of an ion is defined as  $\omega_c = \frac{zB}{m}$  (see Section 1.1), we group a cluster of ions whose mass to charge ratio belong to a small interval of values; i.e.

$$G = \{i \mid i \in [\frac{m_0}{z_0}, \frac{m_1}{z_1}]\} \quad (1)$$

Various properties of an ion group can be computed. The *center of gravity (cog)* is defined as the average position of the group's ions; i.e.  $G_{cog} = \sum_{i=0}^{G_N} pos(i) / G_N$ , in which  $pos(i)$  denotes the position of an ion, and  $G_N$  is the number of ions in the group.

Next, we take the group's *cog* xy coordinates from three subsequent timesteps:  $p(t_{i-1}) = (x_{i-1}, y_{i-1})$ ,  $p(t_i) = (x_i, y_i)$ ,  $p(t_{i+1}) = (x_{i+1}, y_{i+1})$ . If they are not colinear, we can compute the *center of rotation* of the group in the XY plane. We achieve this by solving the system of three equations obtained from the circle equation  $(x_k - x_0)^2 + (y_k - y_0)^2 = r^2$ , by substituting  $k = i - 1, i, i + 1$ . The point  $(x_0, y_0)$  is the *center of rotation*.

The *frequency* of the group is the frequency of the group's *cog* rotating around the center of rotation. The cyclotron phase  $\phi$  of an ion is the angular coordinate in the cylindrical coordinate system placed at the centre of rotation. The *dephase* of the group indicates the difference in cyclotron phase between the outermost ions of a group; i.e. ions at  $\phi_{min}$  and  $\phi_{max}$  in Figure 4.

#### 3.1 Geometrical Icons

A *comet icon* has been developed that encodes the density of the ions in the cluster. The comet icon is chosen since mass spectrometry physicists believe that ions move in a coherent group in which low  $m/z$  values belong to the head and high  $m/z$  values belong to the tail (see Figure 3). A varying density indicates that ions in the cluster are moving with respect to each other, and will be represented by the comet icon changing its shape.

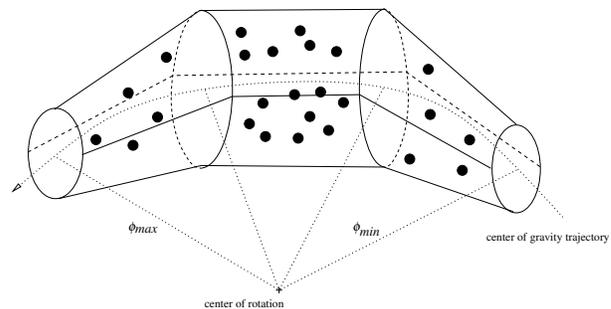


Figure 4: Comet icon construction in 2D: Partition the group in a set of disjunct sections. Each section is represented by a circle, with radius proportional to the included number of ions. The comet is constructed by connecting the circles.

We show how a *comet icon* can be constructed (see Figure 4). First, the group's dephasing property is used to compute the angles

of the outermost ions with respect to the center of rotation of the group ( $\phi_{min}$  and  $\phi_{max}$  in the Figure). Then, the interval  $[\phi_{min}, \phi_{max}]$  is partitioned in a number of disjoint volumes. For each volume a circle is constructed perpendicular to the center of gravity trajectory. The radius of the circle is proportional to the number of ions in the volume; i.e. for partition  $s$ , the radius is  $r_s = n_s \cdot k$  where  $n_s$  is the number of ions in partition  $s$  and  $k$  is a user defined scale factor;  $r_s$  is further clipped to fit in a user defined range  $[r_{min}, r_{max}]$ . The circle is positioned according to the center of gravity of each volume. The comet icon is constructed by connecting the circles, resulting in a tube shaped object. Finally, the comet is scaled along the Z axis, so that its height corresponds to the group's length in the Z direction.

Comet icons encode many useful aspects of the group motion. First, the cyclotron and trapping motion of the group is clearly shown (see point 1 in Section 1.1). Also, the comet shows the dephasing of the group (point 3 in 1.1). Finally, the shape represents the ion density of the group. During animation, changes in the density can be easily monitored.

### 3.2 Frequency Icons

Various icons have been designed that encode ion group motion as phase and frequency information.

The *frequency icon* encodes the difference between the measured group's cyclotron motion frequency and the expected group frequency ( $\omega_c^{theory} - \omega_c^{measured}$  in Point 2 of Section 1.1). The frequency icon is represented as a small circle on an axis. Figure 5 is an example which shows 4 frequency icons. In the normal situation, i.e.  $\omega_c^{theory} - \omega_c^{measured} = 0$ , the circle will be positioned in the origin of the axis. When the frequency perturbs, the circle will be positioned to the left or right of the axis.

The *dephase icon* encodes the dephasing information of the group. The dephase icon is represented as a dial. For each cluster we compute the difference in phase of the cluster's outermost ions, expressed in degrees. We map this value to the angle of the dial. The dephase icon will represent a phase coherent cluster as a thin arc. When the cluster gradually dephases, the disc gradually fills up to a full circle. Four dephase icons are shown on the right side of Figure 5.

Individual frequency icons depict the measured frequency of an ion cluster with respect to the theoretical frequency. In addition, frequency icons can be plotted together on one plot to depict measured frequency of an ion cluster with respect to other ion clusters. This is schematically shown in Figure 5. On the Y-axis are the clusters' mass to charge ratios. On the X-axis are the clusters' frequencies. The theoretical frequency  $\omega_c = \frac{zB}{m}$  is plotted. The frequency icons are positioned along the Y-axis according to the cluster's  $m/z$  value. Phase locking occurs when frequency icons have different  $m/z$  values but equal frequencies. This can be seen easily in the plot.

In summary, a mass spectrometry physicist can gain insight to the frequency perturbations of ion clusters (via frequency icons), cluster dephasing (via dephase icons), and phase locking (via relative icon positions).

### 3.3 Camera control

Camera control is used for analyzing relative motion of an ion cluster. The camera pose is parameterized by ion cluster properties and is positioned at each timestep.

The camera pose is constructed as follows. For a selected cluster we consider the path of its center of gravity. At each time step we compute a Frenet frame along the path; i.e. a coordinate system describing the motion of the center of gravity. The camera is positioned relative to this moving coordinate system. In this way,

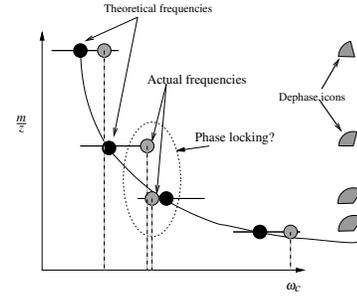


Figure 5: Layout of frequency icons: Frequency icons  $\omega_c^{measured}$  (depicted as small grey circles) are laid out with respect to the theoretical frequency  $\omega_c^{theory}$  (depicted as the monotonic decreasing curve).

the camera can be tethered to the moving cluster. The camera is positioned along the Z-axis or in the XY-plane of the Frenet frame and the field of view is chosen so that all ions of the cluster can be seen (fig. 6).

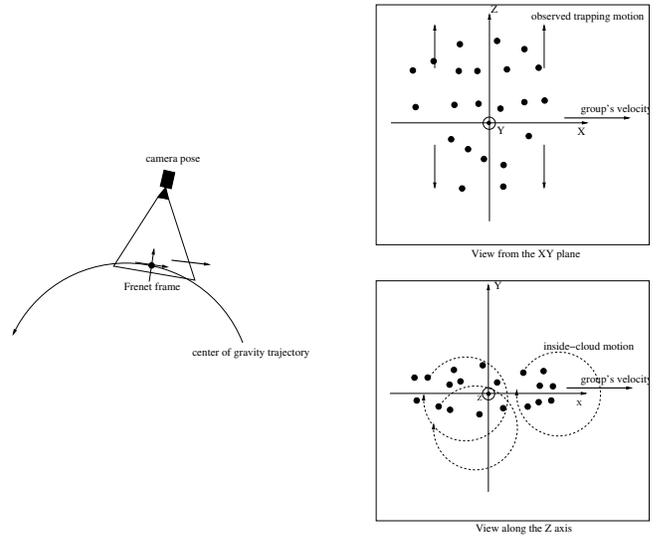


Figure 6: Parameterized camera. Left: the camera is tethered to Frenet frame computed for the center of gravity of an ion cluster. Right top: Relative – inside-cloud – motion of the ions in the cluster can be seen when the camera is positioned along the Z-axis of the Frenet frame. Right bottom: Trapping motion of the ions is viewed when the camera is positioned in the XY-plane of the Frenet frame.

When the camera is positioned along the Z-axis of the Frenet frame, the relative motion of ions inside the cluster can be viewed. Trapping motion can be viewed by positioning the camera in the XY-plane of the Frenet frame. In addition, when the camera is zoomed out, inter-cluster interactions, e.g. cluster collisions, can be observed.

Vectors are rendered to show the direction and magnitude of the motions relative to the camera.

## 4 IMPLEMENTATION

The graphical part of the application is been implemented with the use of VTK library. The construction of VTK pipeline and the graphical interface are implemented in *python* with the help of the wxWidgets library. The computational part of the visualization

(computing the statistical properties) is done in C++ and wrapped as a *python* module.

The system is able to handle medium size FTMS simulations using on a desktop PC with AMD64 CPU, 1 GB memory, 250 GB hard disk and 6600 nvidia graphics. The ion simulations we have visualized involved  $5 \cdot 10^4$  ions simulated over  $10^5$  time steps, resulting in files of approximately 60 GB. A small window of timesteps are cached in main memory, on which statistical properties are computed. As it is now possible to obtain disk capacities of 1TB on a standard desktop PC, simulations involving up to 1 million ions can be visualized with no need for specialized hardware.

When rendering 50k particles as points the system can achieve frame rates of about 70fps.

## 5 RESULTS

The system has been used to visualize FTMS simulations of  $5 \cdot 10^4$  cytochrome C ions with an accurate isotope and charge state distributions. The data are ion  $x, y, z$  coordinates for each timestep and  $\frac{m}{z}$  ratio of each ion. The data file contained  $10^5$  timesteps. Assuming 12 bytes of data per particle ( $x, y, z$  coordinates stored as 4-byte floats) this gives the file of size  $5 \cdot 10^4 \cdot 10^5 \cdot 12B = 60GB$ . The duration of the simulation corresponds to an experimental time scale of a few seconds. This section provides various visualizations of this dataset.

Figure 7 illustrates an ion cluster represented as a comet icon. The thickness indicates the ion density in a given part of the comet. The maximum thickness is set not to exceed the group's boundaries along the Z-axis. The color bar at the top indicates the  $m/z$  distribution of ions: ions with higher  $m/z$  (red color on the bar) are at the comet's tail, whereas ions with smaller  $m/z$  are located in the comet's head. As we can see on the top image, most ions are concentrated in the middle of the cluster. The bottom image shows the same comet icon at different timestep. The shape and size of the icon clearly shows how the ion density in the cluster evolves over time.

Figure 8 shows ion cluster frequencies at two different timesteps represented as frequency icons. The layout of the icons shows how the actual cluster frequencies deviate from the theoretical frequency. On the top image, the clusters seem to be rotating with the frequency close to the theoretical. On the bottom image, however, we can see that some of the clusters have their frequencies perturbed. The reason of those perturbations may be inspected in the 3D perspective view using icons and/or parameterized camera control. The *dephase icons* on the right indicate how the dephasing of ion clusters changes over time.

Figure 9 illustrates the parameterized camera control mechanism. The top image shows a comet icon from the XY plane of the Frenet frame; i.e. trapping motion can be observed. The local coordinate frame is shown in yellow arrows. Small arrows show ion movements relative to the local frame. Red arrows show the direction of trapping motion. The bottom image shows the same comet icon at the same time step from the Z-axis of the Frenet frame.

Finally, Figure 10 shows an overall view of the simulated FTMS cell. The ion clusters are represented as comet icons with  $m/z$  ion distribution illustrated by the color bars over the comets.

## 6 EVALUATION

Fourier transform mass spectrometry is currently playing a leading role in biological mass spectrometry because it demonstrates the highest resolution and mass accuracy. Further improvement of mass accuracy in FTMS is impossible without a better fundamental understanding of the behavior of the ion clouds during the whole cycle of FTMS signal measurement. Novel Particle-in-Cell implementation of the simulations can now ensure realistic conditions by

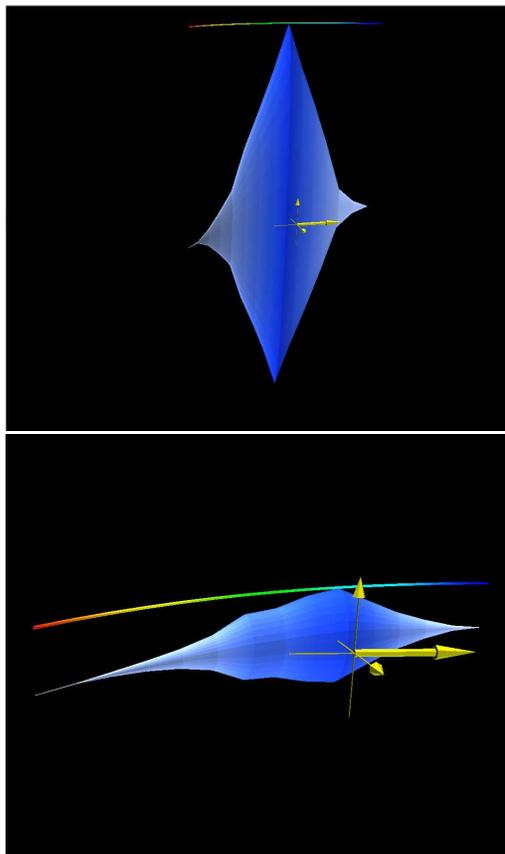


Figure 7: Comet icon. The thickness indicates the ion density in a given part of the comet. The maximum thickness is set not to exceed the group's boundaries along the Z-axis. The color bar at the top indicates the  $m/z$  distribution of ions: ions with higher  $m/z$  (red color on the bar) are in the comet's tail, whereas ions with smaller  $m/z$  are located in the comet's head. Yellow arrows indicate the Frenet frame of the group's *cog*. Top image: Most ions are concentrated in the middle of the comet. Bottom image: The comet at a different time step. The ions are concentrated in the middle of the comet, but due to the trapping motion the ion cluster got flattened. The cluster is now more dephased, which is reflected by the elongated icon shape.

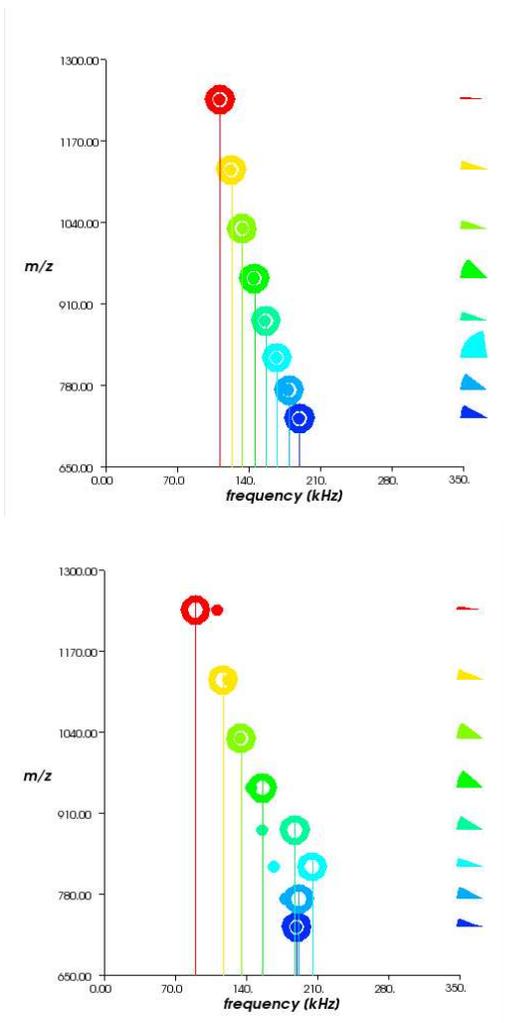


Figure 8: Frequency icons. Several icons are drawn representing the expected and measured frequency for the mass to charge ratio of each ion cluster. Top image: A timestep in which the ion cluster frequencies are very close to the expected frequencies. Bottom image: Ion cluster frequencies deviate from the expected frequencies. This may indicate that cluster-cluster interactions or phase locking of ion clusters has occurred.

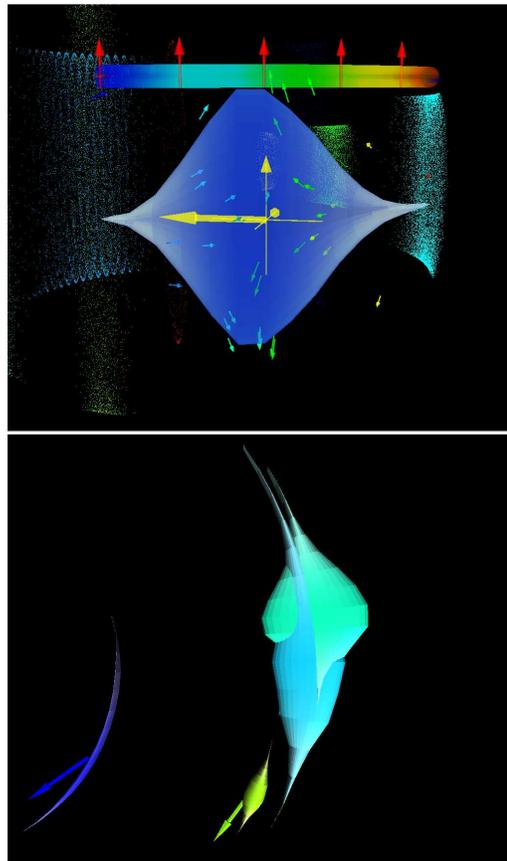


Figure 9: Camera tracing ion cluster represented by comet icon. Top image: View from the XY plane. Visible is the local coordinate frame (yellow). Small arrows give idea of ion movements relative to the local frame. Red arrows show the direction of trapping motion. Bottom image: Camera viewing the traced cluster (blue) along Z axis. Collisions with another ion clusters are clearly seen in this view.

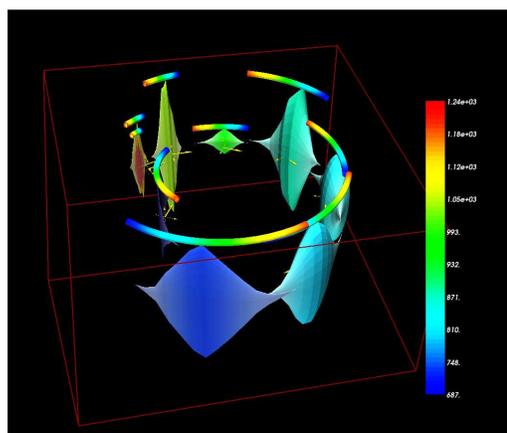


Figure 10: Overall view of the simulated FTMS cell. Multiple ion clusters represented as comets are visible.

taking into account ion-ion interaction, ion-image charge interaction and ion-neutral collisions. Extracting useful information from the data resulting from these simulations is complicated by the fact that the results span multiple physical dimensions such as x,y,z coordinates, time, density and the parameters derived from the basic ion motion such as frequency and phase of the ion cloud.

By using the proposed visualization system, mass spectrometry researchers were able to verify two important hypotheses (see also [1]):

1. FTMS simulations are used to study how the number of ions used in the experiment influences the ion behavior and the measurement accuracy. This is achieved by executing the simulation with different numbers of ions. The researchers found that there are three ranges for the number of ions. Each ion count range results in different ion and ion cloud behavior. For example, when a small number of ions are used, comet like structures appear but dephasing sets in very rapidly. For a moderate number of ions, phase locking of the same  $\frac{m}{z}$  clusters occurs. A large number ions results in phase locking of different  $\frac{m}{z}$  clusters.
2. The excitation profile has also been proven to have considerable influence on ion motions. Simulations are now being used to investigate optimal excitation profiles.

The mass spectrometry researchers were not able to obtain these insights from visualizations with direct rendering and fixed cameras, such as Figure 3. In our case:

- The comet icon encodes the evolution of ion clusters, such as ion density and mass distributions. The dephasing is quantitatively encoded in the shape of the icon. The evolution of dephasing can be monitored by studying the change of icon shape.
- Trapping and cyclotron motions of ions within an ion cluster are studied with the camera control mechanism. Relative ion motions with respect to other ions in the cluster can easily be observed.
- Ion cluster frequency perturbations are clearly visible using frequency icons. The 'normal' (not disturbed) group motion results in a static frequency icon and a static icon layout. When frequency shifts occur, the icon's appearance is modified, attracting user's attention and providing quantitative information on the amount of shift. Slight perturbations of frequency are impossible to comprehend using only a standard 3D view with rotating ion clouds.

## 7 CONCLUSIONS

The goal of this case study was to provide mass spectrometry physicists with high-level visualizations that reflect the way they think about ion dynamics. To do so, we have developed and evaluated a set of visualization idioms that extract high-level semantical concepts from the data and depict these concepts as parameterized geometrical icons.

The way the physicists think about FTMS is twofold. On the one hand they see the particles rotating in the electromagnetic trap. On the other hand they think in terms of frequencies and phases; ions and ion clusters exist only as objects having certain frequency and phase, regardless to their spatial arrangement.

Direct rendering of particle positions is not adequate to obtain insight to the real phenomena that govern the motion of trapped ions. The user is not able to understand the complex interactions of ions and ion clusters based only on the animated images of particles

and their trajectories. Our visualization system presents ion groups by both spatial and frequency icons.

Our system is expected to play crucial role in further development of advanced FTMS simulation techniques. As consequence, the visualization system improves the development and callibration of real FTMS instruments by enabling physicists to better analyze FTMS designs through simulation.

## REFERENCES

- [1] W. Burakiewicz, R. van Liere, E.N. Nikolaev, A.M. Popov, I.M. Taban, and Ron M.A. Heeren. New 3D visualization tools for ion trajectory simulations in FTICR-MS. In *2006 ASMS Conference on Mass Spectrometry*, 2006. Submitted for publication.
- [2] Christopher S. Co, Alex Friedman, David P. Grote, Jean-Luc Vay, E. Wes Bethel, and Kenneth I. Joy. Interactive methods for exploring particle simulation data. Technical report, Lawrence Berkeley National Laboratory, 2004.
- [3] Henk Huitema and Robert van Liere. Interactive visualization of protein dynamics. In *VISUALIZATION '00: Proceedings of the 11th IEEE Visualization 2000 Conference (VIS 2000)*, pages 465–468, Washington, DC, USA, 2000. IEEE Computer Society.
- [4] Patric Ljung, Mark Dieckmann, Niclas Andersson, and Anders Ynnerman. Interactive Visualization of Particle-In-Cell Simulations. In *Proceedings of the 11th IEEE Visualization 2000 Conference (VIS 2000)*. IEEE Computer Society, 2000.
- [5] Kwan-Liu Ma, Greg Schussman, Brett Wilson, Kwok Ko, Ji Qiang, and Robert Ryne. Advanced visualization technology for terascale particle accelerator simulations. In *Proceedings of the 2002 ACM/IEEE conference on Supercomputing*, pages 1–11, Baltimore, Maryland, 2002. IEEE Computer Society Press.
- [6] Alan G. Marshall and Francis R. Verdun. *Fourier Transforms in NMR, Optical, and Mass Spectrometry*. Elsevier, 1980.
- [7] S.E. Parker and R. Samtaney. Tokamak plasma turbulence visualization. In *Proceedings of the 11th IEEE Visualization 1994 Conference*, pages 337–340, Washington, DC, USA, 1994. IEEE Computer Society.
- [8] K.A. Robbins, M. Robinson, and D. Senseman. Visualizing cortical waves and timing from data. In *VISUALIZATION '04: Proceedings of the 11th IEEE Visualization 2004 Conference*, pages 401–408, Washington, DC, USA, 2004. IEEE Computer Society.
- [9] Xiaoming Wei, Arie E. Kaufman, and Timothy J. Hallman. Case study: Visualization of particle track data. In *12th IEEE Visualization 2001 Conference*, 2001.
- [10] Brett Wilson, Kwan-Liu Ma, Ji Qiang, and Robert Ryne. Interactive visualization of particle beams for accelerator design. In *ICCS '02: Proceedings of the International Conference on Computational Science-Part III*, pages 352–361, London, UK, 2002. Springer-Verlag.