



Koltran-Labs Co. Ltd.

Scientific Instrument Services, Inc.



Virtual Device 21.TM Hydrodynamic version



based on CUDA and OpenMP technologies

Moving: Step (X): 10 Step (Y): 10 Step (Z): 10 Clipping step: 1

Virtual Device composition

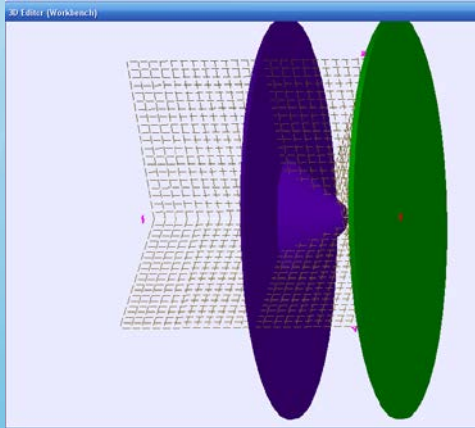
Virtual Device™ 21 is an add-on software package to SIMION® v. 8.1.

Virtual Device is specially designed to enrich *SIMION* with a number of functions, such as fast creation of 3D geometry (CAD program), fast analysis of results of simulation in *SIMION*, and hydrodynamic simulation of gas flow. The main components are

1. **3D editor with OpenGL, for solid geometry development.** This graphical CAD-like program allows the user to compose complex electrode geometries from a set of basic shapes like spheres, cones and bars. The geometry can be exported as a geometry file (GEM) for use in *SIMION*.
2. **3D editor with OpenGL, for hydrodynamics project development.** The 3D editor can also be used for creating hydrodynamic projects (used for calculating gas flows) by specifying boundary conditions for gas flow around/in solid geometries.
3. **Data analysis (for SIMION).** This module performs standard analysis procedures and graphing on *SIMION* data recording results. These include ion distribution, phase distribution in different planes, calculation of histograms of exit parameters (and therefore estimation of peak shape), transformation of time of flight into mass/charge terms, and calculation of mass resolution. Data can be transformed into Excel format. The new version supports multi-event analysis with up to 256 events.
4. **Virtual Jet.** This module calculates gas flow around/in electrodes (hydrodynamics simulation) under conditions of supersonic, compressible, inviscid flow (e.g. supersonic jet or ESI). The hydrodynamics project defined in *3D editor* is loaded into *Virtual Jet* for calculation. The mathematical/physical model of gas flow is based on the *large particle model*. To accelerate gas flow simulation, **CUDA** and **OpenMP** technologies were used. The results of simulation (series of arrays with gas flow distributions) can be exported as files in the *SIMION* PA format.
5. **Collision model and ESI interface.** An essential part of *Virtual Device* is the model of ion collision with a buffer gas, as a Lua program. The combination of the collision model and *arrays with 3D gas flow distribution* (calculated by *Virtual Jet*) allows simulating ion movement in ESI interfaces in *SIMION*.

Logic of investigation by SIMION and Virtual Device 21

“3D editor”

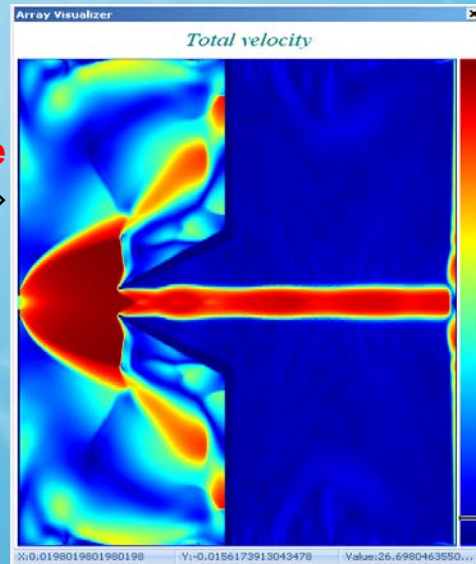


Developing electrode geometry and hydrodynamic model boundary conditions.

GEM or PA files (electrostatic)

VJE file

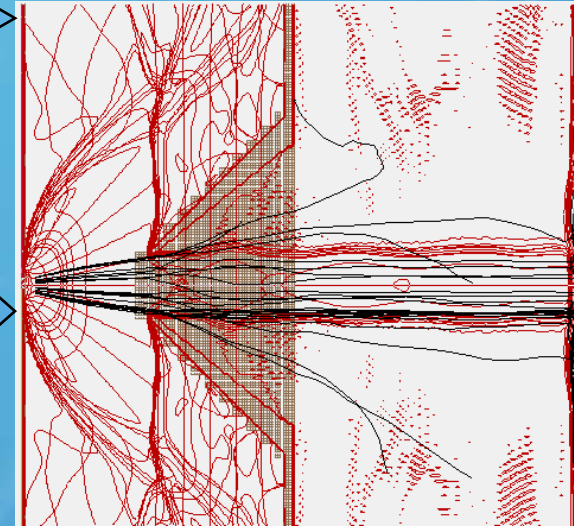
“Virtual Jet”



Hydrodynamic flow solver

PA files (flow)

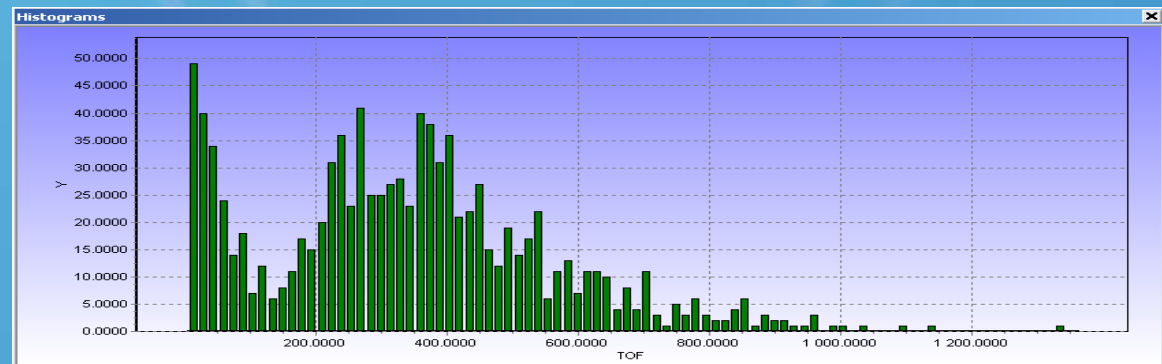
“SIMION 8.1”



Ion movement simulation (with collision model)

“Data Analysis”

Data Recording Output



Analysis of simulation results

3D editor: Developing a SIMION 3D project

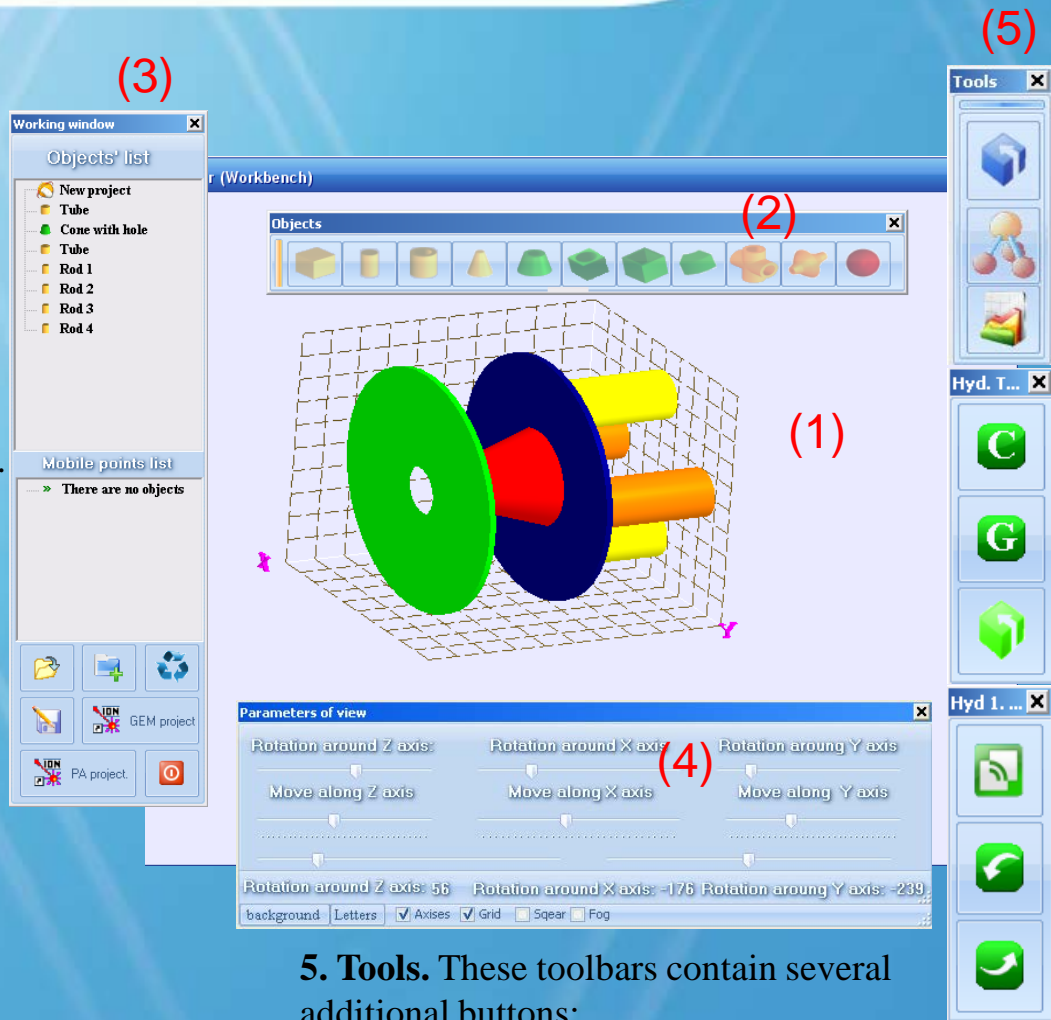
The 3D editor consists of five windows:

1. **3D editor (Workbench).** In this window the user can see the 3D geometry and the grid containing it. The user can rotate the 3D view by dragging the mouse.

2. **Objects.** This toolbar contains icons of the geometrical objects that the user can click on to insert into the project and specify its parameters.

3. **Working window.** This window provides options to create a project, save to or load from a file, transform the project into a SIMION geometry file (GEM), or add or remove geometrical objects in the workbench. This window also contains the tree of the objects and points in the project.

4. **Parameters of view.** This window lets the user move and rotate the 3D view along each of the three axes: X, Y, and Z.



5. **Tools.** These toolbars contain several additional buttons:

- Set SIMION project parameters.
- Set mobile reference points.
- Run scientific calculator.
- Define Hydrodynamics simulation parameters (Hyd. Tools)

3D editor: Developing a hydrodynamic project

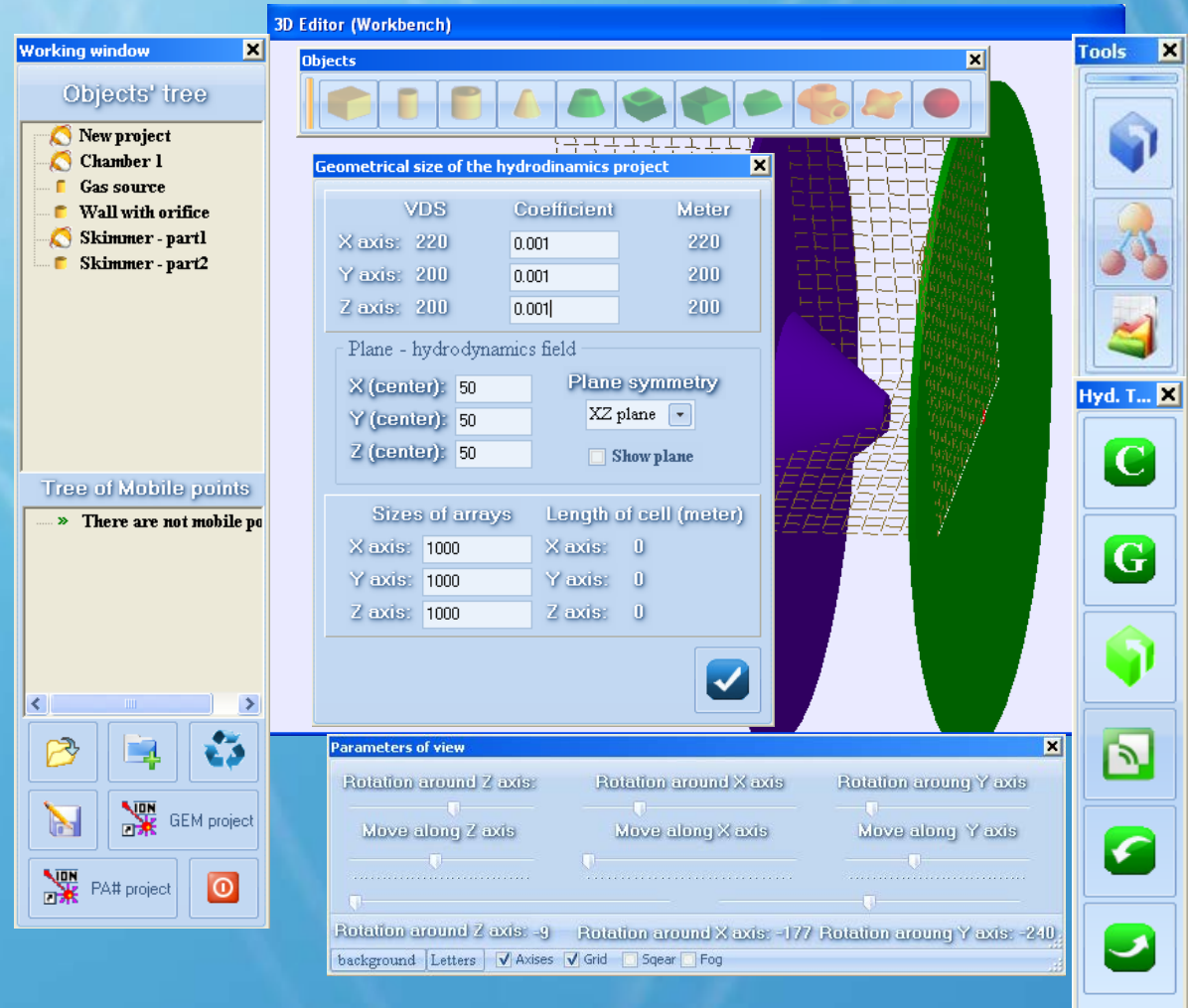
The objects in 3D editor can also define boundary conditions for hydrodynamic (flow) problems. Object types include

1. **Solid object.**
2. **Volume with gas parameters.**
3. **Source of gas.**

Volumes and sources are defined by four additional gas parameters: pressure, temperature, mass and velocity direction.

The combination of solid objects, volumes and gas sources makes the hydrodynamic project (Hyd. project).

To calculate gas flows, the Hyd. project has to be exported into Virtual Jet (VJE file), where the gas-dynamic simulation can be done.

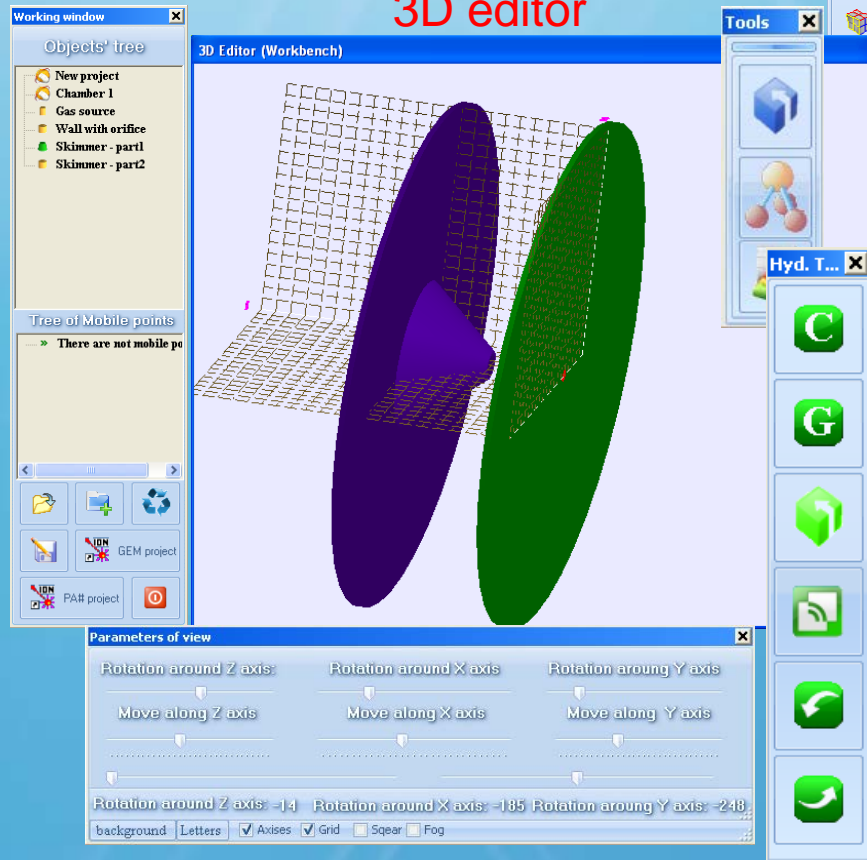


The user can define the sizes of arrays and volumes of a simulation in physical units.

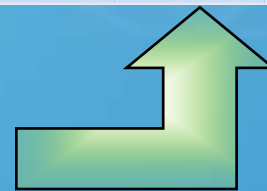
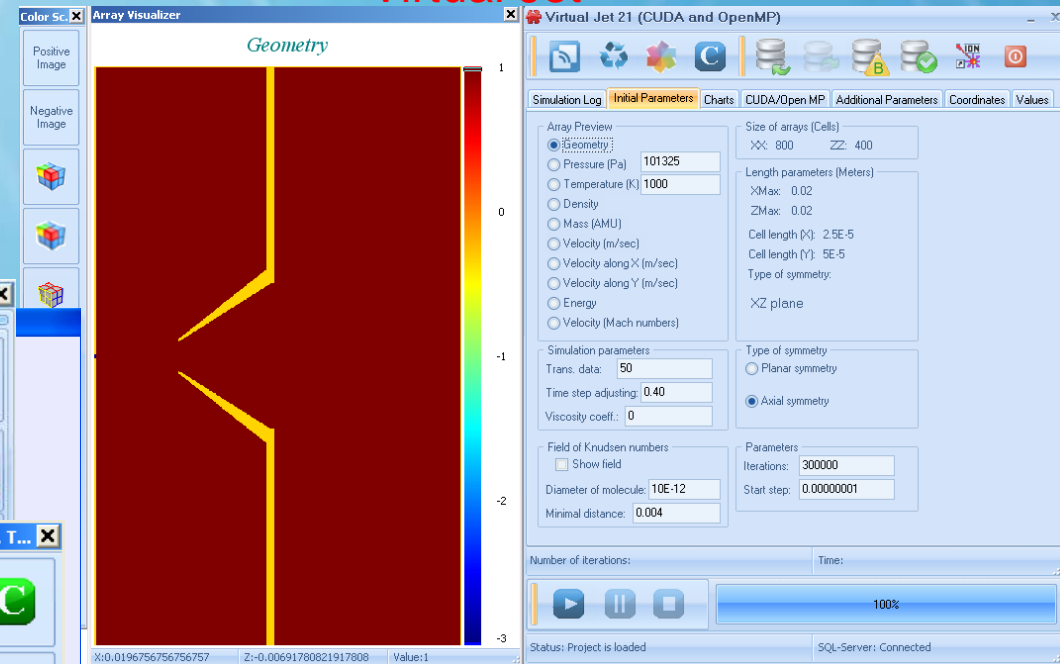
Virtual Jet: Navier-Stokes flow solver

- A hydrodynamic project is a series of arrays with gas and geometry parameters.
- It is developed in 3D editor.
- It has to be exported into Virtual Jet if gas flow simulation is desired.

3D editor



Virtual Jet



- Virtual Jet allows loading hyd. projects from 3D editor.
- Arrays are visualized graphically.
- The user can specify how the arrays are visualized.
- To run a gas flow simulation, simulation parameters should be set.
- Save simulation results in SIMION PA format.

Virtual Jet: Navier-Stokes flow solver

Simulation of gas flow in/around electrodes is based on the **large particle method**. The acceleration of simulation is based on CUDA and OpenMP technologies.

The ideal gas movement can be described by non-stationary Euler equations (in vector notation):

$$\frac{d\rho}{dt} + \text{div}(\rho V) = 0$$

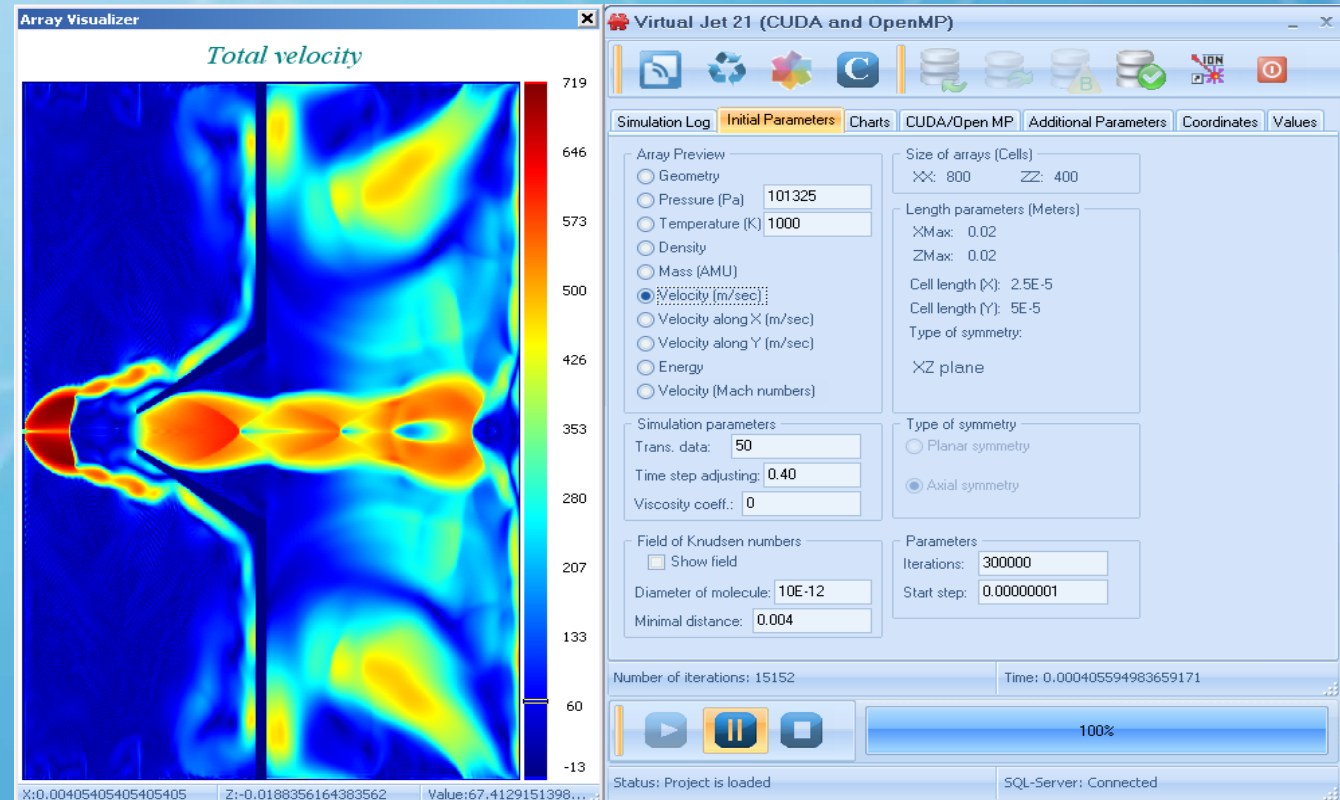
$$\frac{d(\rho v)}{dt} + \text{div}(\rho v V) + \frac{dP}{dy} = 0$$

$$\frac{d(\rho u)}{dt} + \text{div}(\rho u V) + \frac{dP}{dx} = 0$$

$$\frac{d(\rho w)}{dt} + \text{div}(\rho w V) + \frac{dP}{dz} = 0$$

$$\frac{d(\rho E)}{dt} + \text{div}(\rho E V) + \text{div}(P V) = 0$$

ρ – density distribution in time, V – vector of velocity, u, v – projection of V on axis x, y , P – pressure, E – total energy.



The results of simulation are gas flow arrays. These can be visualized as a color pictures. The user monitor the main parameters during the simulation. Data can be stored in a database (MS SQL Server). The sequences of pictures can be exported as AVI files; therefore, the user can easily create a video of gas flow around/in electrodes. The user can also transform the series of arrays into PA files, which can be loaded into the SIMION 8.1 workbench and used in ion movement simulation under electromagnetic forces.

Stability of numerical scheme

1. The stability of a numerical scheme can be evaluated by the following way. We can use Taylor series, which is a representation of a function of sums of terms that are calculated from the values of the function's derivatives at a single point.

$$\frac{d\rho}{dt} + \text{div}(\rho V) = 0 \quad \frac{d(\rho v)}{dt} + \text{div}(\rho v V) + \frac{dP}{dy} = \frac{d}{dx} \left(\rho \varepsilon_x \frac{dv}{dx} \right) + \frac{d}{dy} \left(\rho \varepsilon_y \frac{dv}{dy} \right)$$

$$\frac{d(\rho u)}{dt} + \text{div}(\rho u V) + \frac{dP}{dx} = \frac{d}{dx} \left(\rho \varepsilon_x \frac{du}{dx} \right) + \frac{d}{dy} \left(\rho \varepsilon_y \frac{du}{dy} \right)$$

$$\frac{d(\rho E)}{dt} + \text{div}(\rho E V) + \text{div}(P V) = \frac{d}{dx} \left(\rho \varepsilon_x \frac{dE}{dx} \right) + \frac{d}{dy} \left(\rho \varepsilon_y \frac{dE}{dy} \right)$$

2. The stability of a numerical scheme depends on Courant-Friedrichs-Levy condition (CFL condition).

$$\frac{V \cdot \Delta t}{\Delta} < 1,$$

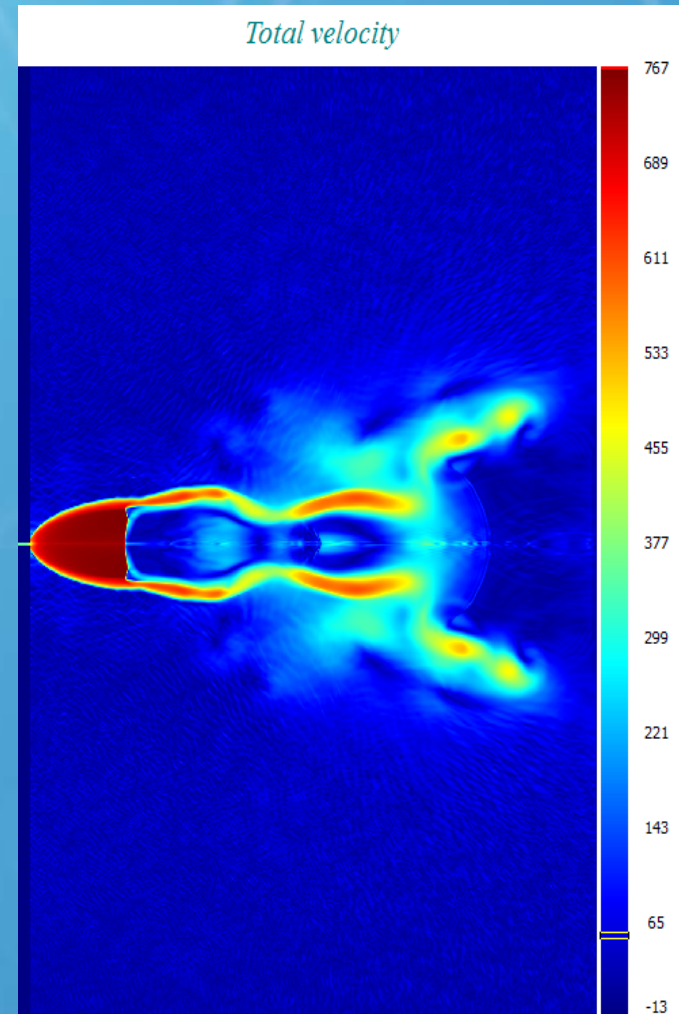
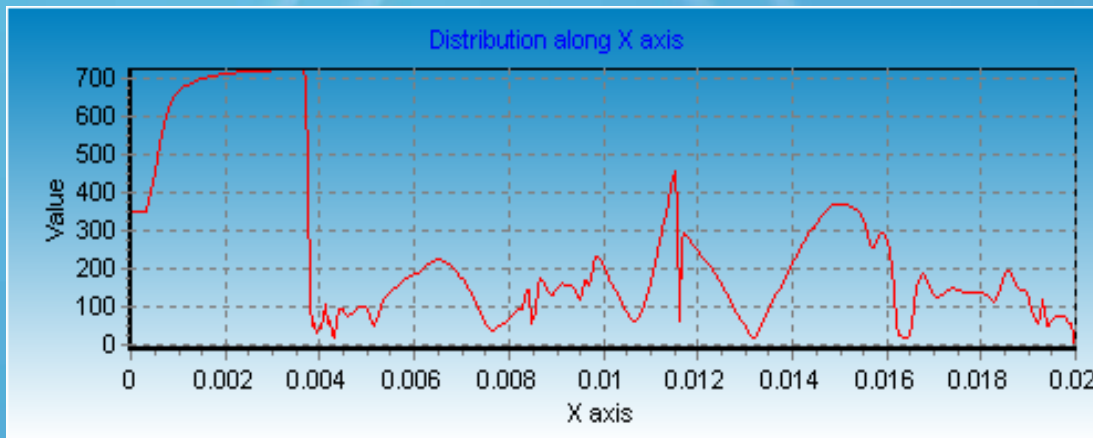
V – velocity(max),

Δt – time step, Δ – size of cell(max)

Tests: Supersonic jet simulation

The **large particle model** has been used for simulation of the supersonic jet in a capillary – skimmer region. The result of this simulation is space distribution of gas parameters like pressure, temperature, velocity, density and so on. The Mach disk position was used as test. The orifice diameter is 0.1 mm. Gas source pressure is 1 atmosphere.

Background pressure (Pa)	Mach disk position Numerical simulation (mm)	Mach disk position Experimental results (mm)
490	1.43	1.46
100	2.14	2.6
50	3.0	3.5
10	6.2	7.08
3	10.0	11.5



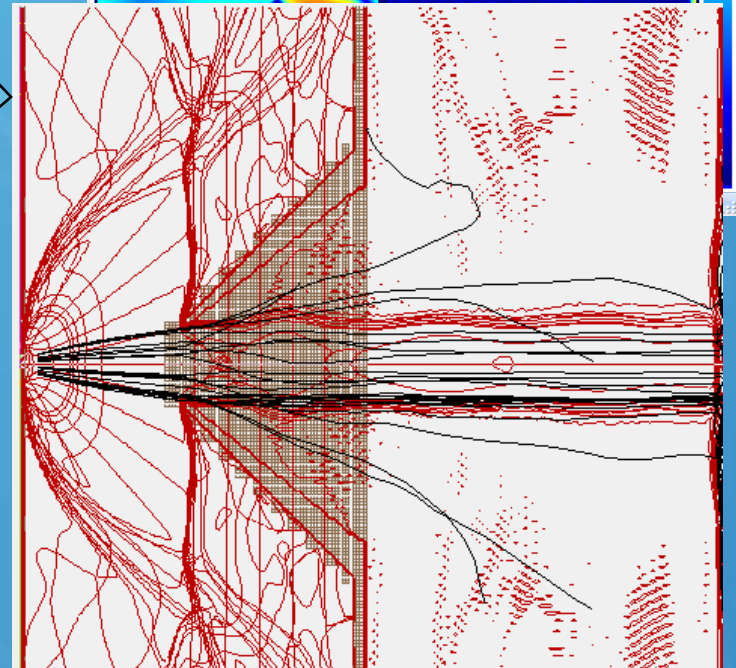
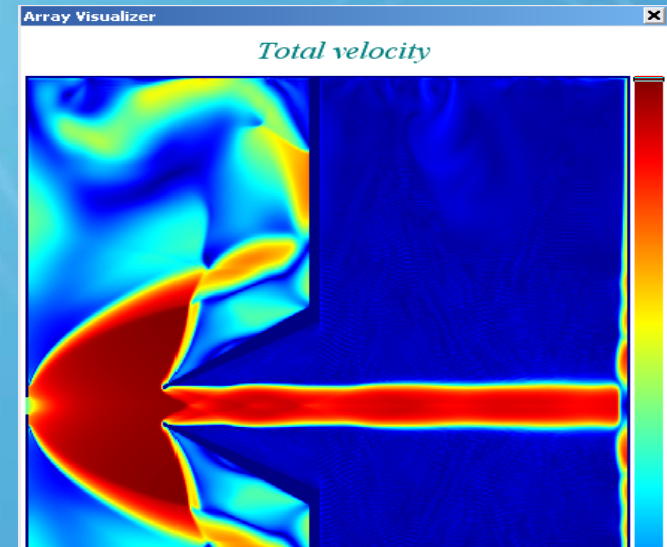
Virtual Jet: exporting data into SIMON 8.1

SIMION 8.1 offers some new capabilities for Lua user programs to read field data directly from PA arrays in memory. Virtual Device can utilize these new capabilities for simulation of ion movement in non-stationary gas flow in the following way.

The user can save the results of a Virtual Jet simulation (series of pressure, temperature, velocity, and density arrays with axial space distribution) as a number of files in PA format. These PA files can be easily added to the SIMION workbench in the usual way.

During ion movement simulation, gas parameters at the local ion position can be read directly from these PA files from the Lua code by just a couple of commands.

The speed of Direct Monte Carlo simulation in SIMION is quite high, which allows simulating ion movement under high pressures like one atmosphere. The combination of Virtual Device 21 and SIMION 8.1 affords the possibility of performing investigation by Direct Monte Carlo simulation in the pressure range from atmosphere down to the free molecular regime.

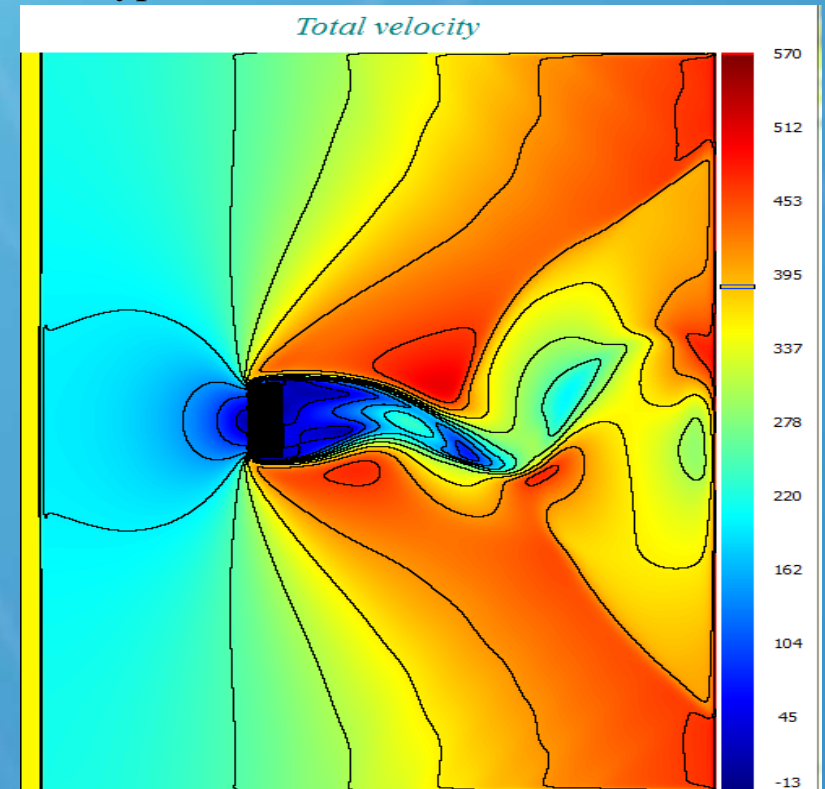
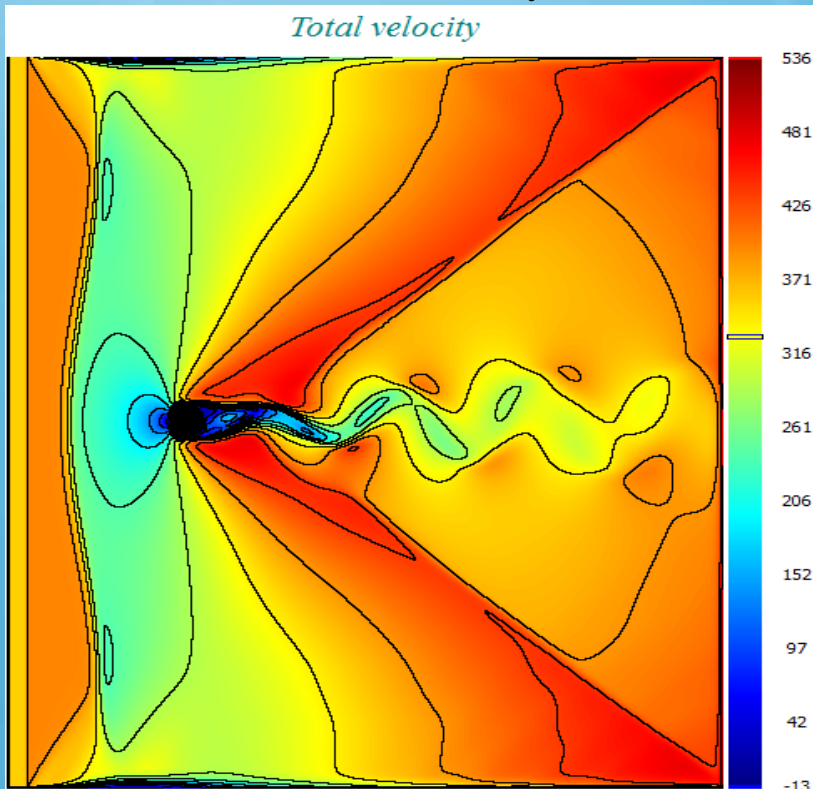


Applications of Large particle model for turbulent simulation

The simulation of turbulence gas flow based on idea that movement can be splitted in two kinds of movements. 1. **Large structure** (ordered whirlwinds). 2. **Small structure** (stochastic fluctuations).

The large whirlwinds can be simulated by using 'Large particle model', where the lowest size of a whirlwind is the size of the cell. This part of the model depends on dissipative terms, which existed in numerical approximation of Navier – Stokes (Euler) equations. Therefore process of energy dissipation is defined by dissipative terms.

Simulation of stochastic fluctuation based on using numerical modeling of solution of Boltzmann equation (Gauss Distribution). Key role in such simulation is hypothesis about molecular chaos.



Simulation of stochastic fluctuations in SIMION

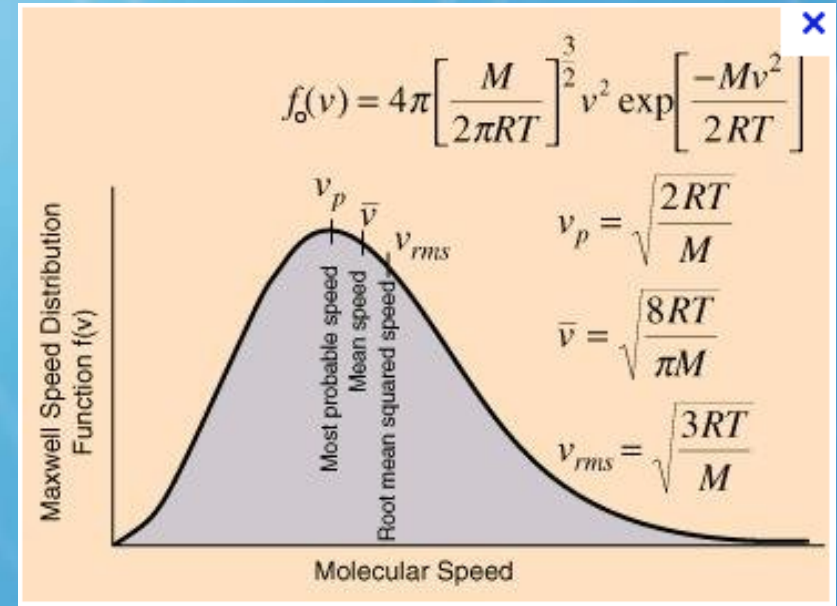
Velocity distribution of gas in SIMION can be presented as following:

$$V = V_{lp} + V_{Mx}$$

Large particle model

Maxwell distribution

$$\frac{df}{\partial t} = n \int_{-\infty}^{\infty} \int_0^{4\pi} (f^* f_1^* - ff_1) \cdot v_r \sigma \cdot d\Omega \cdot dv_1$$



Modeling of stochastic fluctuation based on generation of random direction of velocities of molecules, but those velocities are distributed according to Maxwell distribution. We used Birds' algorithm for numerical modeling of Maxwell distribution.

The resulting velocity of buffer gas molecule is used in simulation of collision.

The usage of V_{lp} and V_{mx} depends on Knudsen number.

Collision model (DSMC): algorithm

The algorithm of ion movement in a gas is realized in Lua code and consists of the following steps:

- Defining the ion position in space.
- Defining gas parameters at the local ion position. Gas parameters can be obtained from PA files (outputted by the Virtual Jet gas flow calculation) plus Maxwell distribution.
- Calculation of distance between collisions.
- Simulation of the collision between the ion and a buffer gas molecule.
- Simulation of ion movement under influence of electrical and magnetic fields.

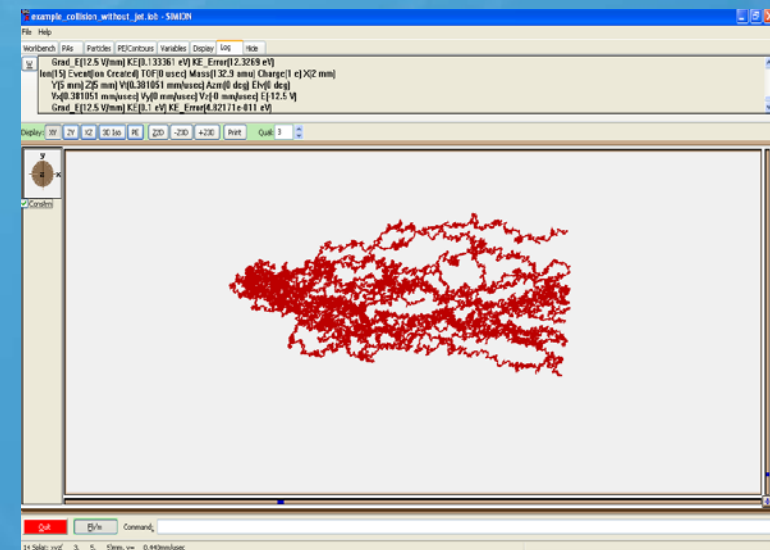
Simulation of ion collisions is based on the hard sphere model, where entrance parameters are mass, velocities of ion and buffer gas molecule. The key features for this model are the value of cross-section and the random number generator.

A high quality generator necessary for calculation of the dispersion vector is provided by SIMION. Therefore the ion velocity vector after collision is a function of the dispersion vector and initial parameters of the ion and buffer gas molecules. This model's description in the *Virtual Device* manual includes a discussion of ion cross-section and its usage.

$$V_1' = \frac{M}{m+M} \cdot V \cdot v + \frac{m}{m+M} \cdot V_1 + \frac{M}{m+M} \cdot V_2$$

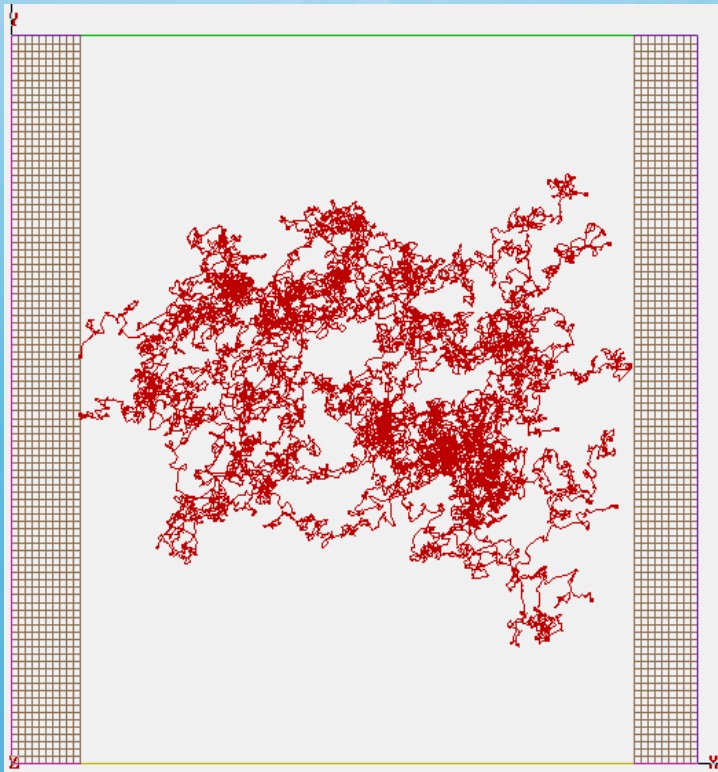
V_1, m - velocity and mass of ion before collision

V_2, M - velocity and mass of buffer gas molecule before collision



Tests: Collision model

A comparison of ion mobility coefficients measured experimentally and to those calculated from Direct Monte Carlo simulation in SIMION are used as tests for the collision model. The geometry in the numerical tests is two parallel plates separated by a distance of 10 mm and potential difference of 1 V. The initial ion energy is 0 eV. The cross-section was calculated according to 'Chapman – Enskog kinetic theory'.



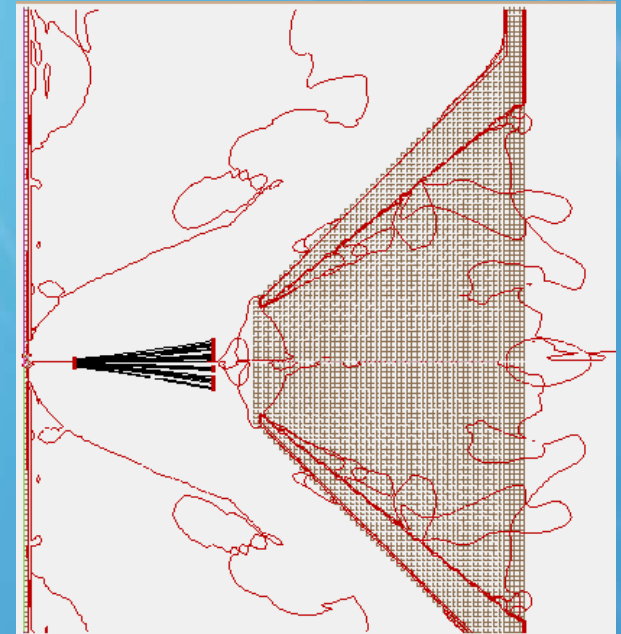
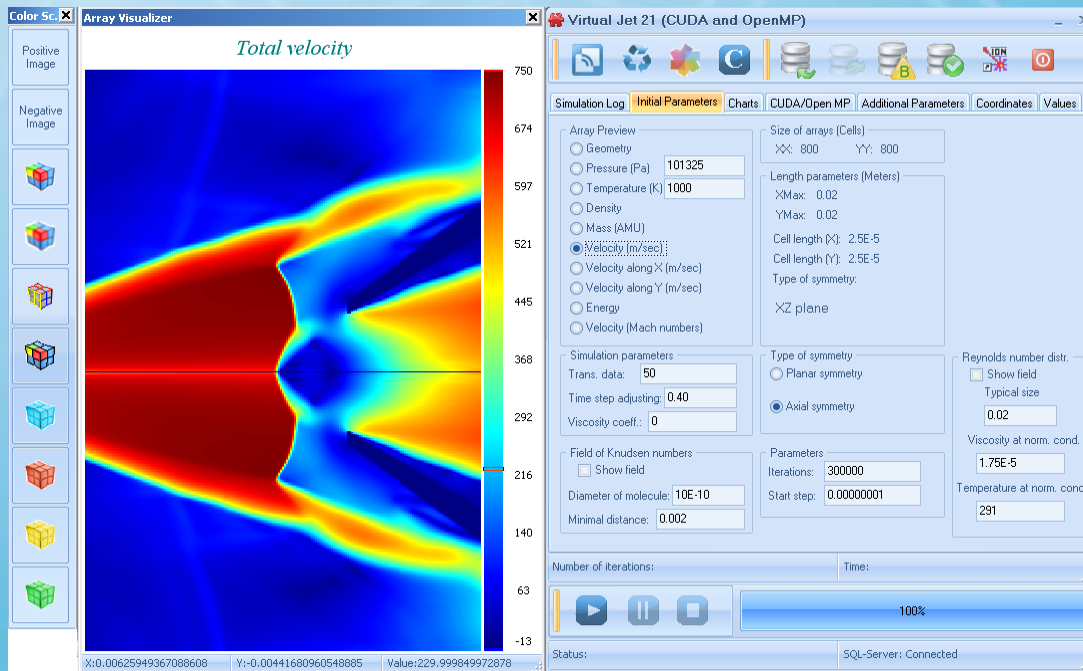
Ion (+)	Mass of ion (amu)	Coefficients of ion mobility, numerical simulation $\frac{cm^2}{B \cdot c}$	Coefficients of ion mobility, experimental results
Cs	132.9	2.16	2.21
Rb	85.468	2.157	2.24
K	39.098	2.5	2.55
Na	22.989	2.75	2.85
Kilpatrick data	244	1.23	1.28
Kilpatrick data	660	0.85	0.845
Kilpatrick data	2122	0.426	0.411

Coefficients of ion mobility are calculated from SIMION simulation according the formula:

$$K = \frac{V_d}{E}$$

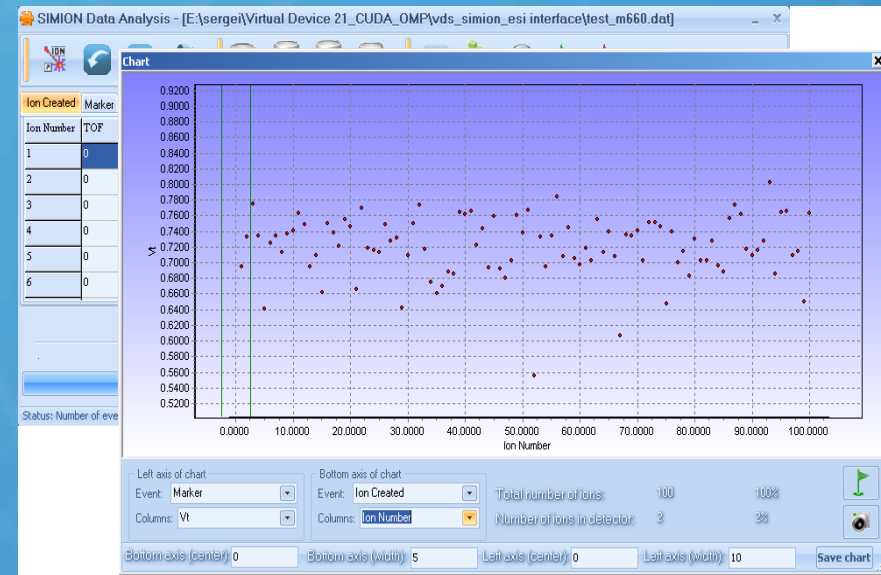
V_d – average velocity of ions,
 E – electrical field

Tests: Movement of ions in region orifice - skimmer



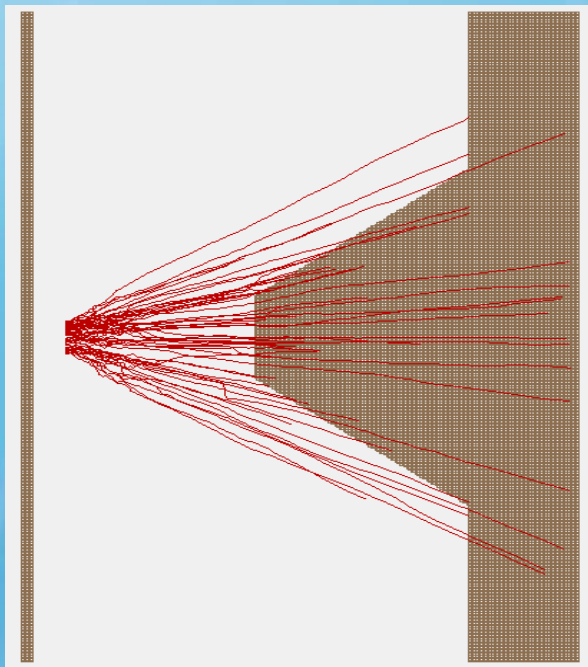
Virtual Jet simulation (Velocity range):
672 m/sec-750m/sec

ION	SIMION simulation (Velocity range)
Cs	640m/sec – 840m/sec
Rb	640m/sec – 840m/sec
Kp data m=660amu	680m/sec – 770m/sec



Example: ion movement in a supersonic jet

An example of ion movement in supersonic jet in an orifice – skimmer region is presented here (simulation in SIMION). This example illustrates the influence of potential on ion movement in a supersonic jet. The simulation shows that the application of potential leads to an increase in ion energy, which becomes higher than any energy that ions can get through collisions. But this is valid only in the mass range $[30 - 2.97 \cdot 10^8 \text{ amu}]$.

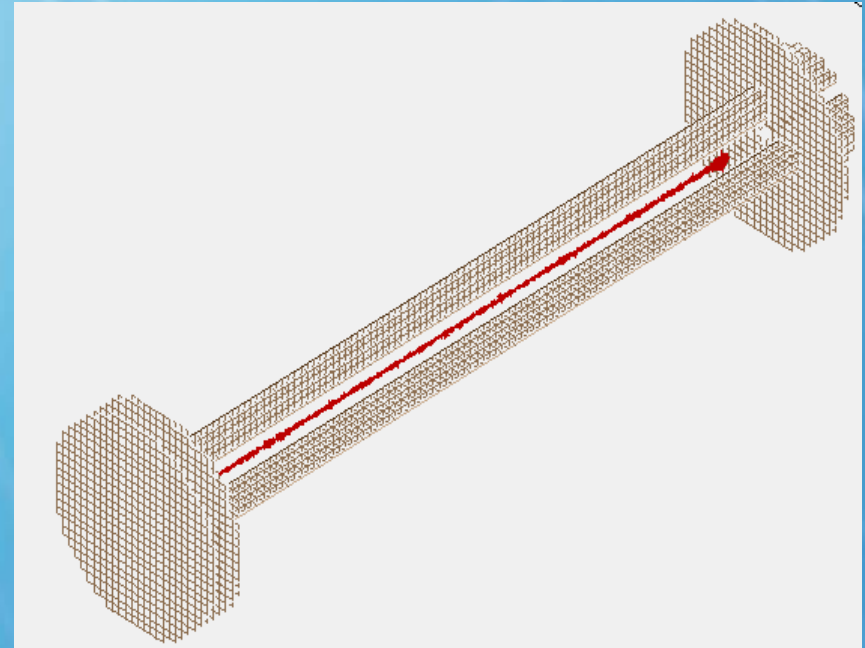
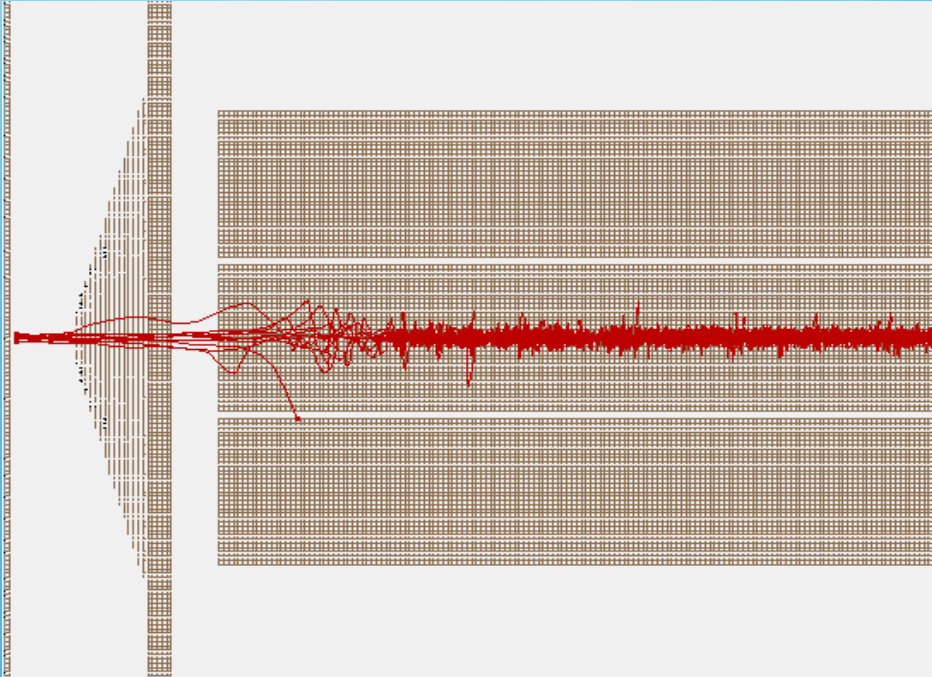


Mass (AMU) Charge +1	Average energy of ions without potentials on electrodes (ev)	Average energy of ions with potentials on electrodes 1kv, (ev)
315	0.841	0.857
788	2.101	2.23
$2.7 \cdot 10^5$	724	728.195
$2.53 \cdot 10^6$	$6.777 \cdot 10^3$	$6.787 \cdot 10^3$
$6.3 \cdot 10^6$	$1.63 \cdot 10^4$	$1.64 \cdot 10^4$
$2.97 \cdot 10^8$	$7.5 \cdot 10^5$	$7.97 \cdot 10^5$
$7.42 \cdot 10^8$	$1.993 \cdot 10^6$	$1.986 \cdot 10^6$
$1.72 \cdot 10^9$	$4.338 \cdot 10^6$	$4.2 \cdot 10^6$
$4.23 \cdot 10^9$	$1.0 \cdot 10^7$	$8.6 \cdot 10^6$
$5.17 \cdot 10^9$	$1.33 \cdot 10^7$	$1.298 \cdot 10^7$
$1.29 \cdot 10^{10}$	$3.1 \cdot 10^7$	$2.89 \cdot 10^7$
$1.15 \cdot 10^{10}$	$2.9 \cdot 10^7$	$2.25 \cdot 10^7$

These results were obtained in a joint project between Institute for Analytical Instrumentation / RAS and Institut für Nukleare Entsorgung Forschungszentrum / Karlsruhe.

Example: ion movement in supersonic jet in skimmer-quad

In this example the region of an orifice-skimmer and the entrance into a quadrupole are characterized by a supersonic jet, where an ion gets significant acceleration by the gas flow via collisions and by the electrical field.

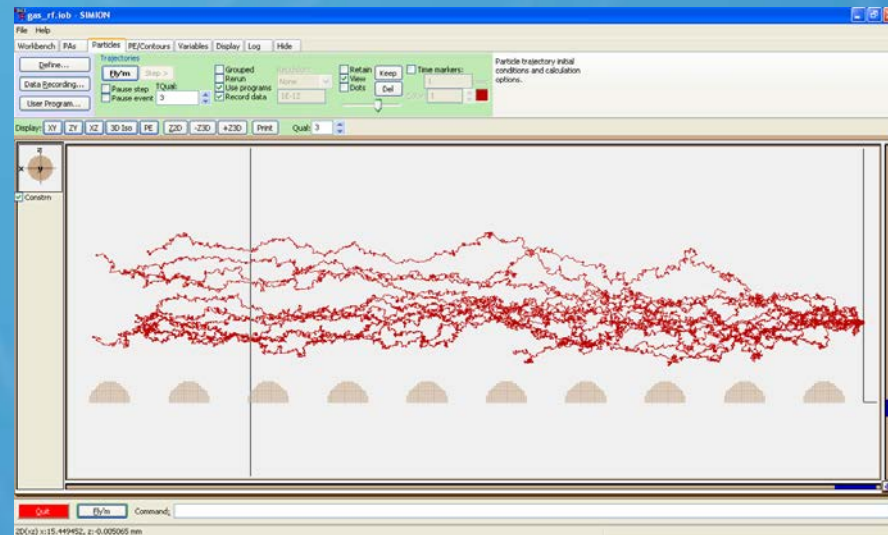
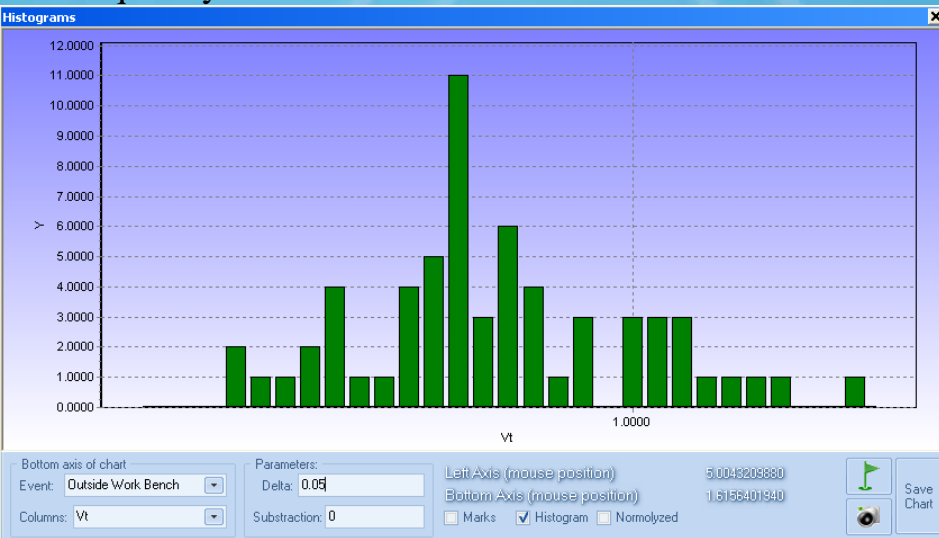
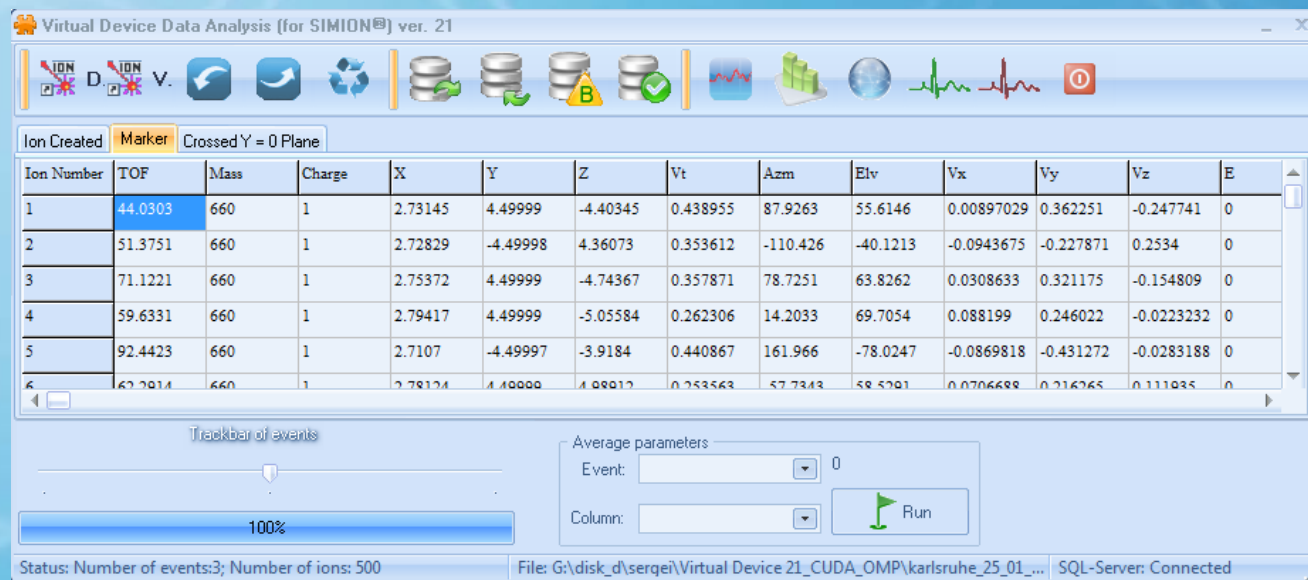


The subsequent quadrupole region is characterized by a resting gas, where the resting gas is simulated as a Boltzmann distribution. The resting gas dampens ions in the quadrupole.

Such approach to simulation combining the large particle model + Boltzmann distribution + collision model allows simulating ion movement from atmosphere to free molecular regime for very complicated geometries. All models are realized in Lua code – an essential part of Virtual Device 21.

Virtual Device Data Analysis (for SIMION)

- Allows loading data from *SIMION* version 8.1.
- Supports multi-event analysis (up to 256 events).
- User can obtain ion mobility coefficients from simulation results.
- Possibility to calculate a spectrum from results of simulation.
- Charts of ion and phase distributions can be quickly created.



CUDA and OpenMP

Gas dynamic simulations are very time consuming, but significant reductions in run-times can be obtained by using special technologies. Virtual jet allows using two main technologies:

- **CUDA** is NVIDIA's parallel computing architecture. It enables dramatic increase in computing performance by harnessing the power of the graphics processing unit (GPU) hardware.
- **OpenMP** (Open Multi-Processing) is an application of programming interface (API) that supports execution of simulation on several CPU cores.

Test 1. OS Windows 7 (32 bit), Intel(R) Core(TM) I7 CPU, 2.2Ghz, 3Gb RAM. 5000 iterations.

Array size	One thread	Open MP (8 threads)	CUDA: GeForce GT540M
array 400x400	5.07 min.	1.30 min.	2.10 min.
array 1000x1000	67 min.	38.28 min.	13.44 min.

Test 2. Computer parameters : OS Windows 7 (32 bit), Intel(R) Core(TM) 2DUO CPU, 2.2Ghz, 3Gb RAM. 5000 iterations.

Array size	CUDA: GeForce GT 240	CUDA: GeForce GT 950
array 400 × 400	2.07 min.	2.41 min.
array 1000 × 1000	12.00 min.	17.0 min.