CWI Quarterly Volume 7 (2) 1994, pp. 115 – 130

Applications of Scientific Visualization in a Physics Environment

Hans J.W. Spoelder*, Frans C.A. Groen *,+

(*) Department of Physics and Astronomy
Free University
De Boelelaan 1081, Amsterdam, The Netherlands

(+) Faculty of Mathematics and Computer Science
University of Amsterdam
Kruislaan 403, Amsterdam, The Netherlands

Scientific visualization is rapidly becoming an indispensable tool for presentation of scientific data. Reaching beyond the indisputable success of presentation of (large) data sets requires among others some form of quantification of the visualization. This together with the possibility to interact with a visualization and to tightly couple it to model calculations are necessary ingredients for successful application of the tool in physics.

For the quantification to a certain extent methods from image processing seem applicable. However, more general methods usable for quantification need to be developed in close cooperation between the visualization tool-maker and tool-user. The same applies to the coupling of simulation and visualization.

As an illustration, some applications realizing aspects of the above mentioned topics with commercially available software, are surveyed.

1. INTRODUCTION

In the introduction of his book on numerical methods HAMMING [1] writes in 1962 that "the purpose of computing is insight, not numbers". The advances in hardware and software since 1962 have dramatically increased the computing power available and thus the ability to produce numbers. However the difficulty of getting insight into the, numeric, results of scientific endeavour seems to have increased equally. This is illustrated by the fact that almost 25 years after Hamming's book appeared, [2] in their National Science Foundation report on "scientific visualization" almost reiterate Hamming when they state that the goal of visualization is "to provide new scientific insight through visual methods". In a follow-up article [3] the same authors stipulate the convergence of the objectives of tool-makers (visualization scientists) and tool-users (scientists and engineers) around - software - visualization environments and define a -hardware - three-tiered hierarchy of visualization facilities: supercomputers, minisupers and advanced workstations.

The hardware and software developments in the past years have solved a number of short-term needs of scientists with often eye-catching results showing that exploitation of the strength of the human visual system [4] can indeed lead to new insight [5], [6]. However the use of these advanced visualization tools has been limited to a relatively small group of users.

Currently the objectives of tool-makers and the tools-users converge around the development and usage of visualization environments running on advanced workstations. The strength of this type of platform is its decentralization, i.e. the general availability on laboratory scale, the weakness, as stressed in [3], is the lack of support staff which makes the user-friendliness of the visualization tool and support for easy integration with user applications all important. To meet these demands the structure of the visual environments has shifted in the past years from monolithic programs to application builders like AVS [7] and DX [8] which allow flexible integration of user-defined software with pre-defined visualization modules.

An example of a demanding group of users of visualization tools are physicists: not only on the scale of 'grand challenges' but also within a laboratory environment. In the latter case the diversity of applications and the large number of applicants puts the usability of tools to a severe test and helps to shape the definition of future software developments. Consequently it pays to survey typical visualization applications within a laboratory.

The usage of the tools can be categorized into three groups: *presentation*, *interaction* and *analysis*.

Presentation focusses on methods and techniques to visually represent numerical data. For scalar data sophisticated visualization techniques are availabale. The proper visual representation of vector fields, which is of pivotal importance to many applications, is still open to debate.

Interaction refers to situations in which the presentation is used as an intermediate between the user and a simulated model system. Essential for this type of work is that the feedback between user and visualization is done in (pseudo) real-time allowing a visual investigation of the effect of model parameter variations. In many physics applications the simulation code for this type of work already exists in the form of (a rusty deck of cards of) Fortran programs. The challenge in that case is to effectively couple visualization and computation with minimal change of the existing simulation code.

Analysis refers to the extraction of information from the visualized data set, the assessment of a model and the subsequent estimation of quantative (model) parameters. Examples of this type of work are found on the border between visualization, more-dimensional image processing and parameter estimation.

In this paper we will survey the usage of visualization tools with respect to the afore mentioned categories and indicate how well they can be realized with the aid of commercially available software packages. To do so we will firstly discuss the visualization infrastructure and then review exemplary applications. Finally we will draw conclusions on future developments for visualization environments. It should be stressed that the work discussed here will bear the mark of a tool-users view on visualization.

2. INFRASTRUCTURE

An abstract representation of the visualization infrastructure used in our laboratory is shown in Figure 1. This realization can be seen as a typical example of general visualization infrastructures [12], [13]. Visualization is based on commercially available software. Three different software packages provide the user with environments for different type of activities: *Scil/Image* is used for image processing (analysis), *Data Explorer (DX)* [8] for presentation, interaction and some analysis, and *Uniras* for the annotational type of two and three dimensional line graphics.

All software can either run on user workstations (SUN/Classic, 24 Mb memory) or on one of the departmental compute servers (IBM RS6000/58H, 128Mb memory) with the exception of the core of DX which always runs on the compute servers. Exchange of information between the various packages is rudimentary: solely a low-level coupling between Scil/Image and DX is possible in the form of – TIFF coded – images.

For single image presentations or small scale animations the local user workstation suffices: (series of) images can be stored in the local memory. Large scale animations require a more sophisticated environment to allow for editing, mixing with sound, real-time viewing and transfer to portable media. The lower part of Figure 1 represents our state-of-the-art realization which is currently being set up. Animation sequences produced by one or more of the software packages result in series of TIFF files [14] which are encoded into a single MPEG file [9]. This MPEG file can be processed in a number of ways. With the use of dedicated hardware it can be viewed in realtime by the user on his local workstation [11]. Editing and mixing with sound can be done with an authoring system [10] resulting in the production of a complete animation. For primary storage we have chosen CD. This medium has clear advantages over the traditional video both in quality and portability. Compared to laserdisc technology it is far more cost effective. The CD can be viewed on any commercially available CDi player. If necessary a video tape copy can be made.

For interaction import of user code in the visualization environment is necessary. Both the visualization package DX and the image processing package Scil/Image have support for this although the integration is certainly non-trivial.

Visualization is demanding both in disk capacity and network bandwidth. Within the laboratory data storage is centralized on dedicated fileservers. Visualization is X11 based thus naturally introducing the client/server concept which concentrates the compute intensive work. The compute servers are interconnected via a fiber link (bandwidth 240 MB/s) whereas the workstations are connected to the servers via UTP category 5 (upto 155 MB/s) point-point connections. This setup makes the bandwidth per scientist optimally scalable and reduces the with visualization associated data movement. Connectivity with extraneous data, for instance stored at SARA (Amsterdam), is currently limited to a bandwidth of 2MB/s thus necessitating local buffering.

Summing up, one can say that the strength of the setup is that each user within the laboratory is capable of using visualization tools close to the place of his actual experiment, although it is difficult to meet the demand for resources like memory, disk storage and CPU power.

3. APPLICATIONS

The boundaries between the various aspects of scientific visualization as mentioned in the introduction are diffuse. Presentation, interaction and analysis can partially be realized with traditional line-oriented graphics. Their merits are beyond dispute and will not further be emphasized here. Rather than using a rigid classification we will give three examples of work currently being carried out and show how visualization is used. The presentation of the examples as far as physical principles involved will be incomplete: only details relevant for the visualization will be introduced.

In Section 3.1 (flow visualization) the focus is on presentation, Section 3.2 (random resistor networks) illustrates the added value of interaction whereas Section 3.3 (remote sensing) elucidates the use of visualization in analysis.

3.1. Flow visualization

The visualization of flow type data is regarded as one the greatest challenges in visualization due to the mixture of vector and scalar data and the large amount of data involved. An important group of flow data is generated by research in transport phenomena which e.g. are at the basis of all kind of environmental research. Application on a different scale of flow analysis of transport phenomena can be found in the field of computer engineering: the design of disk enclosures [15] [1].

The problem can roughly be formulated as: what kind of - steady state - air flow is initiated between two corotating disks, what kind of distortion is induced through the penetrating disk head and what is an optimal enclosure for the disks.

To investigate this, simulation methods are used together with experimental data. For the simulation a finite difference method is used to solve the steady-state Navier-Stokes equations with κ-ϵ turbulence model applied in the core

---

[1] Work on this visualization was initiated during a sabbatical stay of one of the authors at IBM T.J. Watson Research Center, Hawthorne

of the flow and a modified Driest formulation of mixing length in the near wall region. The simulation yields the - axial, radial and tangential - velocity (vector) and pressure (scalar) on a non-uniform, cylindrical, grid. Details are discussed in [15]. Simulations for different geometries of the enclosures and various penetration depths of the head make it possible to chart factors like the power consumption due to viscous dissipation of the airflow, the force and the torque on the actuator due to the shear and pressure drags of the airflow etc.

The compute intensive nature of this simulation work, approximately 3 hours CPU time on a IBM RS6000/590, makes direct interaction impossible. Therefore visualization takes the form of post processing of scalar/vector (pressure and velocity) data on a three dimensional - 128x128x26 - grid.

In Figure 2 the magnitude of the velocity of the air flow around the head of the disk is visualized in the form of a number of isosurfaces chosen equidistantly. Note that due to the lack of support for vector visualization a reduced quantity (the magnitude) is presented. Physically the iso-surface has little meaning, however, it allows easy survey of the vast amount of data showing e.g. the turbulence around the head. This turbulence increases dramatically when the rotational speed of the disk is increased. Investigation of the three dimensional structure of the turbulence induced by the tip of the head is facilitated by exploiting stereo view.

Animation of series of sequences is necessary to detect spatial and/or temporal correlations in the velocity. The effectiveness of the visualization tool could greatly be enhanced when annotational line-graphics, e.g. the boundaries of the discs, could easily be incorporated. Support for this is currently poor. A limitation of this visualization is further that it only deals with scalar fields which makes localization of e.g. turbulences difficult. Arrow like representations of vector field are insufficient. A drawback of the isosurfaces is that they are easily misinterpreted as identical to flow surfaces. The principle of particle tracking could be used to overcome this problem. However this requires sufficient computational power to do so in real-time.

Eye catching though the results may be, it is fair to conclude that the essential activity here is presentation of a large data set facilitating qualitative interpretation of the data. This is useful not only for interpretation of data but also during debugging and program development. When the quantity displayed has a direct physical interpretation (e.g. pressure/concentration) the interpretation of the visualization is facilitated. In most cases the effectiveness of the visualization tool is determined by its ability to easily remap all kind of coordinate systems used for computational purposes and intermix rendering with line-graphics.

The bottle necks are the I/O associated with the large data set involved and the computational power required for rendering. Especially support for a number of users working concurrently with large data-sets is difficult to provide. Interaction and analysis in view of the requirements are difficult to achieve.

3.2. Random resistor networks

When the compute intensive nature of the problem is reduced, or the compute power available is increased, real-time interaction between simulation and visualization, often referred to as computation steering, becomes feasible.

The physics of inhomogeneous media has received much attention over the years [16]. In particular much research effort has been centred around the modelling of superconductors. This and related transport problems in inhomogeneous media can be studied within models of so-called random resistor networks [17]. In such studies the material under investigation is modelled by a two dimensional regular lattice in which the arcs between the nodes represent resistors (see Figure 3). The values for the resistors are determined by some distribution function which is characteristic for the type of problem studied. Specifying a voltage difference between source and drain uniquely determines the system, i.e. determines the unknown voltages on the nodes of the network.

In case of Ohmic resistors, which obey the linear relation $V{=}IR$ with V voltage difference, I the current and R resistor value, the solution can simply be found by solving a sparse system of linear equations [18]. In model studies for superconductors more complex voltage-current relations are used which in some way exhibit the superconductor behaviour that below a certain critical value the resistor value effectively reduces to zero. A typical example of such behaviour [19] is given by

$$\frac{1}{V(i)} = \frac{1}{S\sinh(\frac{i}{i_c})\exp(\frac{-U_{act}}{kT})} + \frac{1}{\rho_f i} \tag{1}$$

Without going into detail we observe that when the second term of eq. (1) becomes dominant the familiar linear relation is found again. The activation energy U_{act} is taken from a distribution function which is dependent on macroscopic parameters like temperature and magnetic field.

The dependency of the overall I-V characteristic of the network on temperature is shown in Figure 4. This average information does not reveal much about local structures in the network. These structures are visualized in Figure 5. The coloured plane represents the state of a network, i.e. voltages, for a given realization, using a linear colour scheme. The solution depends on the distribution function i.e. on the seed of a random generator. To get a visual impression of the specific variance of the solution, i.e. the dependency on the specific realization on the seed of the random series, the user needs to be able to browse through solutions for a number of realizations. In Figure 5 the plane orthogonal to the network realization intersects a number of solutions calculated and draws the isolines found. For zero specific variance, i.e. no dependency on specific realization, these isolines would be straight lines. Thus the visualization gives an impression of the specific variance.

Quantification of the visualization can be achieved for instance by using isosurfaces of constant voltage in different realizations of the network. The area of the isosurface can be used as a measure for the variance. Again parallel planes, i.e. constant minimal area, would indicate no dependency on the specific realization.

For the interpretation of local structure of visualization as shown in Figure 5 the dependency of the solution on physical parameters like temperature is relevant. This work requires, contrary to the foregoing browsing, real interaction. A typical example is the investigation of phenomena like formation and break-down behaviour, i.e. its temperature *(T)* dependence, of superconductors. Using visualization is has been found that depending on the type of I-V characteristic as given in equation (1), temperature dependent clustering of superconducting areas can appear. In some models this clustering plays an essential role [18]. A systematic investigation of the circumstances necessary for this feature of clustering has not yet been undertaken due to lack of appropriate classification tools.

The superconducting areas migrate as a function of time. This dynamic behaviour can experimentally be visualized and measured but also be simulated. The quantitative comparison of both type of data requires image processing operations. To allow the necessary interaction the simulation program as a whole was introduced into the visualization environment, the model parameters being inputs, the resulting voltage vector the output. Using a small C interface the original Fortran source could be used with minor modifications. The advantage of the use of the visualization environment as programming environment is that for the interaction with the input parameters the default interactors of the visualization package can be used. The propagation of the output of the simulation in the environment however is non-trivial.

The visualization in this type of work is predominantly qualitative. However, rather than importing data into a visualization environment, as discussed in Section 3.1, now program parts generating the data are imported.

It is perfectly doable to use the visualization environment more fully as programming environment and split the simulation code into smaller pieces. The advantage is that partial results can easily be visualized, the drawback is that a substantial effort has to be undertaken to import the various program parts. Furthermore this presupposes that the visualization environment has support for essential programming constructs like conditionals and iterative constructs which is only in a limited sense true for DX.

Spatial and temporal correlations are important tools for investigation. Most environments lack tools to flexibly express these correlations or impose constraints upon them.

The reverse problem of determining an optimal set of model parameters given a specific I-V characteristic and the user assisted navigation through the model parameter space is currently under investigation. The quantification of results (part of the analysis) relies heavily on image processing like features as area determination of arbitrary shapes.

3.3. Remote Sensing

The search for characteristics features and the quantification of their behaviour in large data sets can be typed as analysis or modelling. In this case the interaction with the user is vital to test the validity of supposed features. Processing and interpretation of remote sensing data is a good example of this type of work.

The planet Venus is constantly shrouded by a bright veil of unbroken clouds. This seemingly uniform cloud veil does in fact exhibit features in the near ultra violet (see Figure 6). By studying these changing patterns in sufficient detail, one may succeed in understanding the basic physical processes of cloud formation and evolution. Obviously the identity of the aerosols comprising the clouds and the information on horizontal and vertical distributions are essential for such understanding.

An illustration is the simple model for the atmosphere given in Figure 7 [22] which takes absorption and reflection into account. Detailed analysis of earth based observations at various wavelengths have revealed the existence of a thick uniform cloud deck composed predominantly of concentrated sulphuric acid droplets with a narrow size distribution. Limitation of the classical analysis of these type of data is that it is not applied to the entire data set but rather to arbitrarily chosen subsets.

Vast amounts of measurements have become available through the Pioneer Venus Orbiter Cloud Photopolarimeter (OCPP) mission which has measured daily, for almost a decade, the intensity, degree of polarization and angle of polarization at four different wavelengths of the sun light scattered by the Venus atmosphere. Some results have already been published [20, 21]. Handling this massive amount of data, which is exemplary for remote sensing, is still a problem (as is its storage). Remapping the multiple scalar fields onto spherical coordinates and allowing data browsing is a first use of visualization. A typical example of this is shown in Figure 6 where the intensity of the reflected light at two different wavelengths is shown.

A first elementary analysis focuses on the symmetry of this intensity profile. At large wavelengths a highly symmetrical pattern is found indicating the dominance of reflection and homogeneous absorption. Pointwise comparison of the intensity profiles at high wavelength is necessary to support the hypothesis. A direct manipulation of both profiles is not possible since the satellite instrument acquires measurements at different wavelength on different grid points. Thus resampling is a necessary intermediate step. After this, e.g. image processing techniques can be used to study the intensity differences.

An example of this is shown in Figure 8. The lower two figures show the aboslute intensity difference between 365 and 935 nm and 550 and 935 nm, stretched left by a factor of 8 and right by a factor of 64. Clearly the lefthand side shows a more structural difference. This is confirmed by an analysis in which the absolute intensity difference is locally stretched followed by the application of a gradient like operator. The result of this is shown in the upper two figures. The bands shown indicate transition in value. The difference at higher wavelengths (upper right) is small and uncorrelated, whereas the difference at lower wavelengths (upper left) is correlated and larger. This is a starting point for a more elaborate analysis.

Firstly the model parameters can be varied to produce an optimal global fit. Numerical models for this type of light scattering simulation have been developed and the feasibility to integrate them with the visualization environment is currently investigated.

Programming tools to quantify the difference between simulation and measurement and to calculate an optimal search direction are currently lacking in many visualization environment. Secondly, models for the local variations of the absorbance predicting wind velocities, directions etc. can be tested by tracking features in the intensity profile. For this type of work the presented data are an extreme example since the interval between successive measurements is (too) large compared to the wind speed in the Venus atmosphere. Due to this, straightforward automated definition and relocation of 'region of interests' as tracking mechanism for features in the clouds fails. However, correlation between the different wavelengths of selected regions of interest might be helpful in this context.

Quantification of the original data requires a number of image processing like operations (region of interest definition, relocation of patterns, correlation of images) which are slowly finding their way in visualization environments. This quantification is essential for analysis.

The massive amount of data involved poses a number of additional problems like browsing and real-time transformation which are difficult to solve, especially in the context of traditional visualization packages. The simultaneous display of multiple scalar fields (different measurements) enhances the understanding of the data but is still open to debate.

However one should emphasize that analysis of the remote sensing data is almost unthinkable without visualization tools.

4. CONCLUSIONS

Scientific visualization is becoming an indispensable tool for scientific research. The decentralisation of this tool which is possible due to the advances in hardware has certainly contributed to its success. This in itself makes clear that visualization need not be limited to supercomputer applications. A major limitation for use within physical applications is the limited support for vector visualization.

For interaction purposes coupling with existing programs is essential and the current state of tools to do so can be qualified as minimal. Analysis requires a kind of merger between visualization and image processing techniques i.e. methods for quantification of features. A number of techniques like length, area and volume determination are now finding their way into the visualization packages.

All in all this indicates the need to integrate visualization into a more general programming environment. The use of visualization during program development as debugging tool is a good example of the usefulness of this. Viewing the state of current commercially available visualization environments still a substantial support staff is necessary to make it work even on a laboratory scale.

ACKNOWLEDGEMENTS
Many have contributed to parts of the work presented here. The authors would especially like to mention: A.H. Ullings (hardware), R. Wolfre jr. (airflow visualization), G. Doornbos and R.J. Wijngaarde (resistor networks) and W.J. Knibbe (remote sensing).

REFERENCES

1. R.W. HAMMING (1962). *Numerical methods for scientist and engineers*, McGraw-Hill, New York.
2. Figure B.H. MCCORMICK, T.A. DEFANTI and M.D. BROWN eds., (1987). Visualization in scientific computing, *Computer Graphics*, Vol. **21**, No. 6.
3. T.A. DEFANTI, M.D. BROWN and B.H. MCCORMICK (1989). Visualization: expanding scientific and engineering research opportunities, *IEEE Computer*, pp. 13 - 22.
4. E.R. TUFTE (1983). *The visual display of quantative information*, Graphics Press, Cheshire Connecticut.
5. G.M. NIELSON and B. SHRIVER eds. (1990). *Visualization in scientific computing*, IEEE Computer Society Press, Los Alamitos.
6. K.W. BRODLIE ET AL. eds. (1992). *Scientific visualization*, Springer Verlag, Berlin.
7. C. UPSON (1989). The Application Visualization System: a computational environment for scientific visualization, *IEEE Computer Graphics and Applications*, pp. 30-42.
8. B. LUCAS, G. ABRAM, N. COLLINS, D. EPSTEIN, D. GRESH and K. MCAULIFFE (1992). An architecture for a scientific visualization system, *Proceedings of IEEE Visualization 92*, Boston, Ma., pp. 107-114.
9. CCITT RECOMMENDATION MPEG-1 (1993). *Coded representation of picture, audio and Multimedia/Hypermedia Information*, ISO/IEC 11172, Geneve, Switzerland.
10. L.A. ROW K. PATEL, B.C. SMITH and K.LIU (1994). *MPeg Video in software: representation, transmission and playback*, High speed networking and multimedia computing, IS&T/SPIE, Symp. on Elec. Imaging Sci. & Techn., San Jose, CA.
11. MPEGTool: An X windows based MPEG encoder and statistics tool, *Proceedings of the ACM Multimedia*, Anaheim, CA, 1993.
12. J.J. VAN WIJK (1992). Scientific visualization, in P. DEWILDE and J. VANDEWALLE (eds.), *Computer systems and software Engineering state-of-the-art*, Kluwer Academic Publishers, Dordrecht, The Netherlands.
13. L.J. ROSENBLUM (1989). Scientific visualization at research laboratories, *IEEE Computer*, pp. 69-101.

14. *TIFF Revision 6.0 standard*, Alus Developers Desk, Alus, Corporation, Seattly, WA, 1992
15. C.J. CHANG and H. LI (1990). Three dimensional numerical simulation of flows for corotating disks, *Proceedings of the ASME International Computer in Engineering Conference*, Boston, Mass, Vol. **2**, pp. 387-392.
16. *Proceedings of the second international conference on electrical and optical properties of inhomogeneous media*, Physica A, **157**, (1), 1989.
17. D. STAUFFER (1990). *Introduction to percolation theory*, Taylor and Francis, London.
18. E.L. HINRICHSEN (1990). The conductor-superconductor transition in disordered superconducting materials, Physica C, 167, 433-455.
19. G. DOORNBOS (1992). *Numerical and Experimental studies of current-voltage characteristics of Type II superconductors*, Free University, Amsterdam, Internal report.
20. K. KAWABATA, D.L. COFFEEN, J.E. HANSEN, W.A. LANE, M. SATO and L.D. TRAVIS (1980). Cloud and haze properties from Pioneer Venus polarimetry, *J. of. Geophysical Res.*, Vol **85**, No. A13, pp. 8129-8140.
21. W.B. ROSSOW, A.D. DEL GENIO and T. EICHLER, Cloud-tracked winds from Pioneer Venus images, *J. Atmos Sc.*, Vol. **47**, No. 11, pp. 2053 - 2084.
22. W.M.F. WAUBEN (1993). *Multiple scattering of polarized radiation in planetary atmospheres*, Thesis Free University, Amsterdam.

FIGURE 1. Schematic representation of the visualization infrastructure at the faculty of Physics and Astronomy of the Free University. Visualization software can run on local workstations (SUN/Classic) or departmental compute servers (IBM RS6000/58H).

FIGURE 3. Symbolic representation of a random resistor network. The arcs between the nodes represent connection whose resistance value is determined by some model function. Given source and drain voltage the voltages at the nodes can be evaluated.

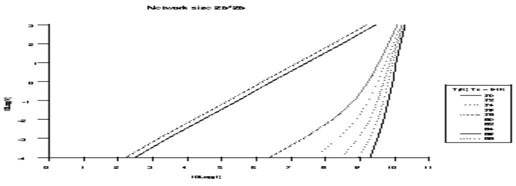

FIGURE 4. Overall I-V response of a random resistor network for different temperature values. The size of the network is 25 square. Note that both the x axis (I) and y axis (V) are logarithmic.

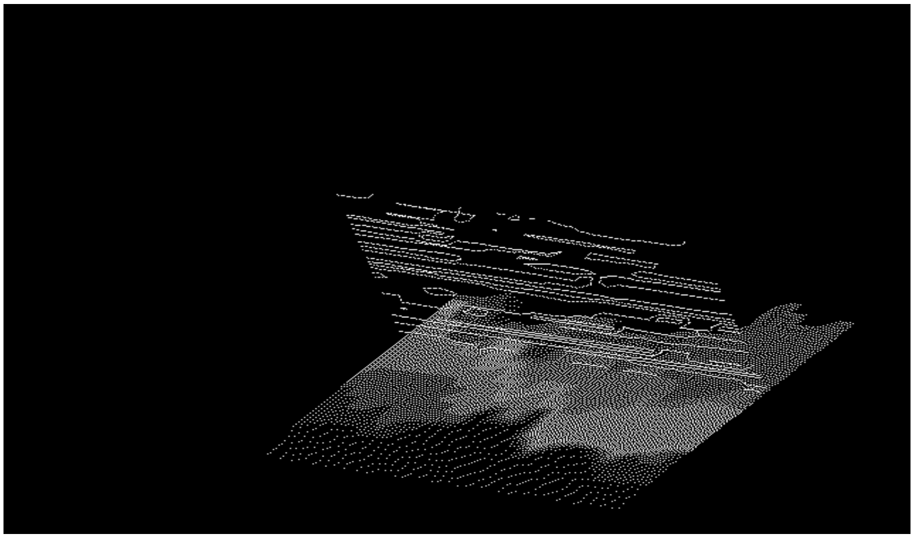

FIGURE 5. Visualization of the state of one random resistor network (colour). The orthogonal plane intersects all realizations calculated and shows isocontours for the voltage value in this plane.

FIGURE 7. Schematic picture of an atmospheric model of Venus. The relative concentration of gas and haze particles and their absorption are model parameters.

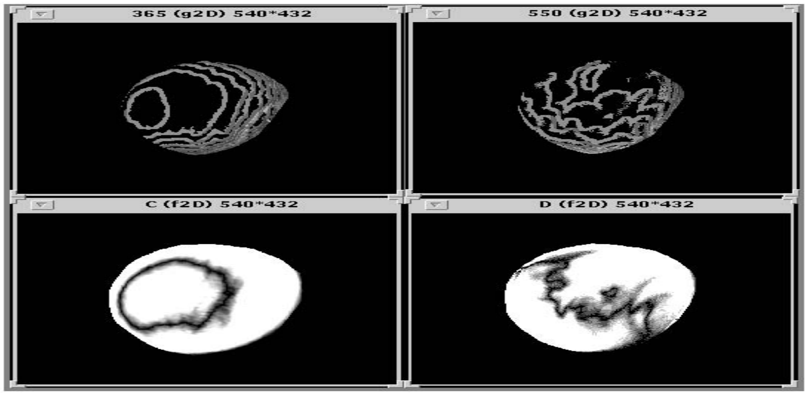

FIGURE 8. Analysis of intensity profiles of Venus atmosphere. Lowerleft $|I_{365} - I_{935}|$ stretched by a factor of 8, lowerright $|I_{550} - I_{935}|$ stretched by a factor of 64. Upperleft and upperright denote transitions in locally stretched intensity of the lower row. Note the random character of the right hand image versus the structured nature of the left hand side.