51

Grupo de Física Matemática
Universidade de Lisboa

FERMILAB
FEB 25 1997
LIBRARY

IFM-5-96

Topics in computational physics(*)

R. Vilela Mendes

Fermilab Library
0 1160 0064638 4

NAME	I.D.No.	M.S.
~~P. Bhat~~	~~8445~~	~~357~~

RETURN TO FERMILAB LIBRARY

(*)Lectures at the Euroconference "*Supercomputation in nonlinear and disordered systems: Algorithms, applications and architectures*", Escorial, September 1996

IFM 5/96

TOPICS IN COMPUTATIONAL PHYSICS

R. VILELA MENDES
Grupo de Física-Matemática
Complexo II - Univ. de Lisboa
Av. Gama Pinto, 2, 1699 Lisboa Codex - Portugal

Contents

1 Introduction **2**

2 Boundary layer control. A problem in fluid dynamics **3**
2.1 *The boundary layer equations* 5
2.2 *Scaling solutions* 7
2.3 *Numerical results* 11

3 Stochastic ground state simulations **13**
3.1 *Quantum Monte-Carlo and the sign problem* 13
3.2 *Ground state processes* 16

4 Handling non-Gaussian data. Characteristic function and large deviations **20**
4.1 *Non-Gaussian data and the neural computation of the characteristic function* 20
4.2 *Large deviation analysis of experimental data* 24
4.2.1 *A summary of large deviation theory* 25
4.2.2 *Large deviation analysis of experimental data* 30

5 A method for fast track recognition in high-energy colliders **32**

6 Adaptive Monte Carlo integration **34**

1 Introduction

Computational Physics is, in a sense, like social science or politics. Social distortions cannot be solved by just throwing money at the problems. Likewise, in computational physics, very seldom can a problem be satisfactorily solved by just flooding it with GFLOPS. In both cases a careful formulation of the problem, a qualitative understanding and, at least, a good qualitative solution should be found before the large investment or the heavy numerics stage is reached.

In computational physics one starts from a **physical problem** for which there is some general **mathematical model.** By mathematical model I mean a set of differential equations, an operator relation in Hilbert space, etc. If the mathematical model cannot be solved exactly, an **algorithm** must be developed for the numerical or symbolic evaluation of the model. Notice however that, in general, an algorithm derived from the mathematical model solves not only our original physical problem but also all the physical problems compatible with the same mathematical model. Therefore, instead of using a general purpose algorithm, it is often convenient to devote some time to make the algorithm specific to the problem at hand. This has a good payoff in terms of computer time, simplicity and reliability of the algorithm.

Once we have decided on an algorithm, the last step is to put it on a **computing machine.** This step should also be taken with extreme care. Do not forget that, because of the finite memory, computers are finite-state machines, not Turing machines. For example: because of discretization no computer can ever be equivalent to a continuous differential equation and there are well-known examples where the discrete-time version is qualitatively different from the continuous one. Therefore the results coming from the computer should always be carefully controlled. This control may take the form of a convergence proof, an analytical error estimate or the agreement of the numerical results with known analytical solutions for some limit value of the parameters.

In Sects. 2 and 3 of these lectures I will describe two computational problems which are handled following this methodology. The first concerns the problem of boundary layer control in fluid dynamics, a problem of great concern in aviation and submarine propulsion and the second deals with attempts to avoid the minus-sign problem in Quantum Monte-Carlo simulations involving fermions.

The important computational lesson to be learned from the boundary layer study is that, by specializing the fluid equations to the concrete physical problem, a considerable simplification is obtained. Then, instead of having to deal with time-consuming general purpose finite-element codes, results of comparable precision are obtained with simple and fast finite-difference implicit algorithms. Furthermore, the prior derivation of approximate analytical solutions, provides a good control on the numerical results and simple design formulas.

For the minus sign problem in quantum Monte-Carlo the lesson to retain is that it is basically a representation problem. Therefore it cannot solved by

improving statistics with faster computers. The solution lies in finding a representation that minimizes the effect or that eliminates it altogether.

A second point that I would like to emphasize is that the role of the computational physicist is not just to solve physical problems that require long or sophisticated computational techniques. It is also to develop (and make available) efficient tools which other scientists may use in their day to day numerical work. This concerns not only new tools for theoretical calculations, as discussed in Sect. 6, but also the development of efficient algorithms for the analysis of experimental data. Let us not forget that Natural Sciences are experimental sciences and, in the hard job of decoding Nature, every bit of help is welcome. This is illustrated in Sections 4 and 5, which deal with methods to analyze non-Gaussian data and algorithms for particle track recognition.

Last but not least, let me point out that the kind of approach towards computational physics, that is advocated in Sects. 2 and 3, has as a side benefit the effect of democratizing computational physics, in the sense that many problems which by brute force methods are only accessible in the fastest parallel supercomputers, become feasible with far more modest computer means. This does not mean that high performance computing (HPC)[1] is not needed. Much to the contrary, the efforts to improve computer hardware and software should be vigorously pursued. What I mean is that the new developments in HPC, coupled with a careful and economical approach to computational physics, will allow us to solve even more difficult problems.

2 Boundary layer control. A problem in fluid dynamics

Whether a flow is laminar or turbulent, the effects of the viscosity of the fluid are greatest in regions close to solid boundaries. The region close to the boundary, in which the velocity varies from zero, relative to the surface, up to its full value, is called the *boundary layer*. The concept of boundary layer introduced by Prandtl in 1904, was a most significant advance in fluid dynamics, in the sense that it simplified the study by separating the flow in two parts: (1) the region where velocity gradients are large enough to produce appreciable viscous forces - the boundary layer itself - and (2) the external region where viscous forces are negligible compared to other forces. From the computational point of view the concept of boundary layer also plays a significant role because, rather than having to deal with time-consuming general purpose finite-element codes, results of comparable precision may be obtained by fast and relatively simple finite-difference implicit algorithms.

When a fluid flows past a solid body, an airfoil for example, a *laminar boundary layer* develops, in general only for a very small distance near the leading edge,

followed by a transition to a *turbulent boundary layer.* Nevertheless, because near the solid wall velocity fluctuations must die out, below the turbulent region there always is a *laminar sub-layer* which in general is very small (of the order of a micrometer). The transition from the laminar to the turbulent region is controlled by the *local Reynolds number,* defined in terms of the effective width of the boundary layer. It also depends on the smoothness of the surface and on the external perturbations. The *skin friction drag* is proportional to the gradient of the longitudinal velocity at the solid boundary. Because of the mixing properties of the turbulent layer, the gradient in the laminar sub-layer is much greater than the gradient at a fully laminar layer. Therefore transition from a laminar to a turbulent layer greatly increases the skin friction drag.

Because of the very large ratio between laminar and turbulent skin friction drag, much effort has been devoted to develop techniques to delay the transition as a means of decreasing fuel consumption and noise. Some of the active control techniques that have been proposed include suction of slow-moving fluid through slots or a porous surface, use of compliant walls and wall cooling (or wall heating for liquids). Injection of fast-moving fluid, on the other hand, is effective in avoiding separation but it increases turbulence. Most of these ideas are fairly old (see for example [2],[3]) however, in view of their interest for the applications and to obtain a more accurate characterization of the physical mechanisms, studies of boundary layer control using these aerodynamic methods are still, at present, being vigorously pursued (see for example [4] [5] [6] [7] and papers in [8])

Another class of techniques for active boundary layer control consists in acting on the flow by means of electromagnetic forces. Here different techniques should be envisaged according to whether the fluid is weakly conducting (an electrolyte like seawater or an ionized gas) or a good conductor (like a liquid metal). Proposals for boundary layer control by electromagnetic forces are also relatively old and trace its origin at least to the papers of Gailitis and Lielausis[9], Tsinober and Shtern[10] and Moffat[11] in the sixties. Interest in these techniques has revived in recent years and some more accurate calculations and experimental verifications have been carried out, mostly in the context of electrolyte fluids[12] [13].

Here I will be concerned with the flow of air along an airfoil when a layer of ionized air is created on the boundary layer region. I will assume that a stream of ionized air is injected through a backwards facing slot placed slightly behind the stagnation point (Fig.1) and the body force acting in the ionized fluid is a longitudinal (along the flow) electric field created by a series of plate electrodes transversal to the flow and placed inside the airfoil with the edges on the airfoil surface. The control of the boundary layer profile, to delay the laminar to turbulent transition, is in this example the physical problem.

2.1 *The boundary layer equations*

Orthogonal curvilinear coordinates are used, with $\tilde{x}$ parallel to the surface along the flow and $\tilde{y}$ normal to the surface. If $\kappa\delta$ is small (κ denoting the curvature and δ the boundary layer width) the conservation and momentum equations in the incompressible fluid approximation may be written

$$\frac{\partial \tilde{u}}{\partial \tilde{x}} + \frac{\partial \tilde{v}}{\partial \tilde{y}} = 0 \tag{1}$$

$$\tilde{u}\frac{\partial \tilde{u}}{\partial \tilde{x}} + \tilde{v}\frac{\partial \tilde{u}}{\partial \tilde{y}} = -\frac{1}{\tilde{\rho}_m}\frac{\partial \tilde{p}}{\partial \tilde{x}} + \tilde{\nu}\left(\frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2} + \frac{\partial^2 \tilde{u}}{\partial \tilde{y}^2}\right) + \frac{1}{\tilde{\rho}_m}\tilde{\sigma}_e(\tilde{x},\tilde{y})\tilde{E}_x(\tilde{x},\tilde{y}) \tag{2}$$

$$\tilde{u}\frac{\partial \tilde{v}}{\partial \tilde{x}} + \tilde{v}\frac{\partial \tilde{v}}{\partial \tilde{y}} = -\frac{1}{\tilde{\rho}_m}\frac{\partial \tilde{p}}{\partial \tilde{y}} + \tilde{\nu}\left(\frac{\partial^2 \tilde{v}}{\partial \tilde{x}^2} + \frac{\partial^2 \tilde{v}}{\partial \tilde{y}^2}\right) + \frac{1}{\tilde{\rho}_m}\tilde{\sigma}_e(\tilde{x},\tilde{y})\tilde{E}_y(\tilde{x},\tilde{y}) \tag{3}$$

$\tilde{u}$ and $\tilde{v}$ are the components of the fluid velocity field along the $\tilde{x}$ and $\tilde{y}$ directions. The tilde denotes quantities in physical dimensions to be distinguished from the adimensional quantities defined below. Eqs.(1-3) are the mathematical model for our problem.

Consider now typical values L_r, δ_r, U_r, ρ_r,ν_r,σ_r, E_r as reference values for, respectively, the airfoil width, the boundary layer width, the fluid velocity, the fluid mass density, the kinematic viscosity, the fluid charge density and the electric field. Then one may define adimensional quantities

$$x = \frac{\tilde{x}}{L_r}\ ,\ y = \frac{\tilde{y}}{\delta_r}\ ,\ u = \frac{\tilde{u}}{U_r}\ ,\ v = \frac{\tilde{v}L_r}{U_r\delta_r} \tag{4}$$

$$\rho_m = \frac{\tilde{\rho}_m}{\rho_r}\ ,\ p = \frac{\tilde{p}}{\rho_r U_r^2}\ , R_L = \frac{U_r L_r}{\nu_r} \tag{5}$$

$$\nu = \frac{\tilde{\nu}}{\nu_r}\ ,\ \sigma = \frac{\tilde{\sigma}}{\sigma_r}\ ,\ E = \frac{\tilde{E}}{E_r} \tag{6}$$

In general $R_L >> 1$. Neglecting terms of order $\frac{1}{R_L}$ and $\frac{\delta_r^2}{L_r^2}$ one obtains

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{7}$$

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho_m}\frac{\partial p}{\partial x} + \nu\omega\frac{\partial^2 u}{\partial y^2} + \gamma\frac{1}{\rho_m}\sigma(x,y)E_x(x,y) \tag{8}$$

$$\frac{\partial p}{\partial y} = \frac{\delta_r}{L_r}\gamma\sigma(x,y)E_y(x,y) \tag{9}$$

where $\omega = \frac{L_r^2}{\delta_r^2 R_L} = \frac{L_r \nu_r}{\delta_r^2 U_r}$ and $\gamma = \frac{L_r \sigma_r E_r}{U_r^2 \rho_r}$. Unless the electric field component normal to the airfoil is very large, one has $\frac{\partial p}{\partial y} \approx 0$ and the pressure term in the second equation may be expressed in terms of the fluid velocity u_e far away from the airfoil

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = u_e\frac{\partial u_e}{\partial x} + \nu\omega\frac{\partial^2 u}{\partial y^2} + \gamma\frac{1}{\rho_m}\sigma(x,y)E_x(x,y) \tag{10}$$

To take into account turbulence effects one should also replace in (10) the velocity fields u and v by $u + u'$ and $v + v'$, u' and v' being fluctuation fields with zero mean, $\overline{u'} = 0$, $\overline{v'} = 0$. The effect of the turbulent field on the mean flow is now obtained by taking mean values. In a two-dimensional turbulent boundary layer the dominant eddy stress is $\overline{-u'v'}$. Assuming the eddy shear stress $\overline{-u'v'}$ and the mean rate of strain $\frac{\partial u}{\partial y}$ to be linearly related

$$\overline{-u'v'} = \epsilon\frac{\partial u}{\partial y} \tag{11}$$

one obtains finally

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = u_e\frac{\partial u_e}{\partial x} + \nu\omega\frac{\partial}{\partial y}\left(\beta\frac{\partial u}{\partial y}\right) + \gamma\frac{1}{\rho_m}\sigma(x,y)E_x(x,y) \tag{12}$$

with

$$\beta = 1 + \frac{\epsilon}{\omega} \tag{13}$$

being, in general, a function of y through the dependence of the eddy viscosity ϵ on the local velocity field. β should be obtained from a turbulence model.

To analyze the scaling solutions and for the numerical calculations of Sect.2.3 define a stream function ψ and make the following change of variables

$$\eta = \left(\frac{u_e}{\nu\omega}\right)^{\frac{1}{2}}\frac{y}{\xi(x)} \tag{14}$$

$$\psi = (u_e\nu\omega)^{\frac{1}{2}}\,\xi(x)f(x,\eta) \tag{15}$$

$$u = \frac{\partial \psi}{\partial y} \ , \ v = -\frac{\partial \psi}{\partial x} \tag{16}$$

The continuity equation (7) is automatically satisfied by (16) and one is left with

$$\frac{\partial}{\partial \eta}\left(\beta \frac{\partial^2 f}{\partial \eta^2}\right) + \xi \frac{\partial \xi}{\partial x} f \frac{\partial^2 f}{\partial \eta^2} + \xi^2 \frac{\partial u_e}{\partial x} + \xi^2 \left(\frac{\partial^2 f}{\partial \eta^2}\frac{\partial f}{\partial x} - \frac{\partial f}{\partial \eta}\frac{\partial^2 f}{\partial \eta \partial x}\right) = -\frac{\gamma}{u_e \rho_m} \xi^2(x)\sigma(x,\eta)E_x(x,\eta) \tag{17}$$

This is now the only equation to be solved. It provides an accurate description of the boundary layer dynamics and, of course, is simpler than the full Navier-Stokes equation. It is therefore for this equation that an **algorithm** will be developed below. It turns out that it may be reliably solved by an implicit finite-difference method.

Before dealing with the numerical solution of Eq.(17), approximate analytical solutions are obtained[15], which will serve as a **control** for the numerical algorithm. Also, approximate analytical solutions provide useful design formulas, something that numerical results cannot do.

2.2 *Scaling solutions*

We assume the electric field to be created by a series of plate electrodes along the z-direction, that is, transversal to the fluid flow. For this electrode geometry the mean electric field in the x-direction may be parametrized by

$$\overline{E_x} = g(x)\frac{l(x)}{l^2(x) + y^2} \tag{18}$$

where x and y are the adimensional coordinates defined in (4). $E_0 = \frac{g(x)}{l(x)}$ is the field at $y = 0$, controlled by the potential differences between the electrodes, and $l(x)$ is of the order of the electrode spacing. For thin (laminar) boundary layers the field E_x may, with good approximation, be considered to be independent of y throughout the boundary layer width, as long as the appropriate charge density profile is chosen (see below).

Ionized air is injected through a slot near the leading edge of the airfoil, being then carried along the airfoil surface by the flow. Because of this injection method the charge density is maximum at the airfoil surface, decreasing to zero at a distance that depends on the fluid dynamics and the injection regime. The dynamically-dependent charge density profile may be parametrized by the formula

$$\sigma(x,y) = \sigma_0\left(1 - \frac{u}{u_e}\right) \tag{19}$$

A scaling solution of (17) is one for which f is only a function of η. Eq.(17) becomes

$$\left(\beta f''\right)' + \xi\,\dot{\xi}\,ff'' + \xi^2\frac{\partial u_e}{\partial x} = -\frac{\gamma}{\rho_m}\xi^2(x)\sigma_0\left(1-f'\right)g(x)\frac{l(x)}{u_e l^2(x)+\omega\xi^2(x)\eta^2} \quad (20)$$

with boundary conditions

$$f(0) = f'(0) = 0 \quad f'(\infty) = 1 \quad (21)$$

where, for simplicity, I have denoted $f' \equiv \frac{\partial f}{\partial \eta}$ and $\dot{\xi} = \frac{\partial \xi}{\partial x}$.

Let the pressure be approximately constant for length scales L of the order of the airfoil, that is $\frac{\partial u_e}{\partial x} \approx 0$. Let also β be a constant. This is the case for the laminar part of the boundary layer. Then the factorized nature of Eq.(20) implies that solutions exist only if

$$\xi^{-2}(x) = \frac{2}{c_1}\frac{\dot{\xi}(x)}{\xi(x)} = \frac{1}{c_3}\frac{g(x)}{\xi(x)} = c_4 l^{-2}(x) \quad (22)$$

c_1, c_3 and c_4 being constants. Therefore

$$\begin{aligned} \xi(x) &= \sqrt{c_1 x + c_2} \\ g(x) &= \frac{c_3}{\xi(x)} \\ l(x) &= \sqrt{c_4}\,\xi(x) \end{aligned} \quad (23)$$

There are two physically interesting situations. The one with $c_1 \neq 0$ $c_2 = 0$ and the one with $c_1 = 0$ $c_2 \neq 0$. The first one corresponds to a boundary layer starting at $x = 0$ and growing with $x^{\frac{1}{2}}$ and the second to a constant width boundary layer. The first one corresponds to an equation

$$f'''(\eta) + \frac{1}{2}f(\eta)f''(\eta) + \left(1-f'(\eta)\right)\frac{\varphi_1}{\varphi_2^2+\eta^2} = 0 \quad (24)$$

with $c_1 = \beta$, $c_2 = 0$, $\varphi_1 = \frac{\gamma\sigma_0 c_3\sqrt{c_4}}{\beta\rho_m\nu\omega}$, $\varphi_2 = \sqrt{\frac{u_e c_4}{\nu\omega}}$ and the second to

$$f'''(\eta) + \left(1-f'(\eta)\right)\frac{a}{b^2+\eta^2} = 0 \quad (25)$$

with $c_1 = 0$, $c_2 \neq 0$, $a = \frac{\gamma\sigma_0 c_3\sqrt{c_4}}{\beta\rho_m\nu\omega}$, $b = \sqrt{\frac{u_e c_4}{\nu\omega}}$.

In the first case one chooses $c_2 = 0$ to obtain a boundary layer starting at $x = 0$. The scaling hypothesis requires then an electric field that is singular at $x = 0$, $y = 0$ ($E_x \sim x^{-1}$). In any case this electric field solution is not very interesting for our purposes because it leads to a boundary layer growth of $x^{\frac{1}{2}}$, as in the free force Blasius solution. Therefore it will be more interesting to consider

a small field free region in the leading edge of the airfoil and match the Blasius solution there with the constant width solution of Eq.(25).

Gailitis and Lielausis[9] have also obtained a theoretical solution of constant width. However they consider a different force field distribution and no dependence of the fluid charge density on the boundary layer dynamics. Therefore their boundary layer profile has a very different behavior.

The solution of Eq.(25) is easily obtained by numerical integration[15]. Notice however that with the replacement

$$\phi(\eta) = 1 - f'(\eta) \tag{26}$$

and choosing $c_4 = \frac{\nu\omega}{u_e}$, which is a simple rescaling of ξ, Eq.(25) becomes the zero-eigenvalue problem for a Schrödinger equation in the potential $a/(1+\eta^2)$,

$$-\phi''(\eta) + \frac{a}{1+\eta^2}\phi(\eta) = 0 \tag{27}$$

One may use the well-known WKB approximation to obtain

$$f'(\eta) = 1 - \frac{(1+\eta^2)^{\frac{1}{4}}}{\left(\eta + \sqrt{1+\eta^2}\right)^{\sqrt{a}}} \tag{28}$$

Eq.(28) is a very good approximation to the exact solution for $a \geq 1$. Fig.2 shows the effective boundary layer width as a function of a. The effective boundary layer width δ^* is defined here as the value of η at which the velocity u reaches 0.95 of its asymptotic value u_e. A very fast thinning of the boundary layer is obtained (several orders of magnitude) for a relatively short range of the a parameter. Fig.2 shows the variation of δ^* for small a. For large a (and small δ^*) one has the asymptotic formula

$$\delta^* \simeq \frac{2.9957}{\sqrt{a}}$$

which is obtained from Eq.(28).

If the longitudinal electric field E_x is assumed to be a constant (E_0) throughout the boundary layer width, with the same charge profile, the solution is even simpler, namely

$$f'(\eta) = 1 - e^{-\eta\sqrt{h}} \tag{29}$$

with $\xi = \sqrt{c_2}$ and

$$h = \frac{\gamma c_2 \sigma_0 E_0}{\beta u_e \rho_\infty} \tag{30}$$

Again, since ξ is a constant, this is not fully realistic because it leads to a constant width boundary layer.

For reference values of the physical quantities in Eqs.(4-6) we take

$$\begin{aligned} U_r &= 100 \text{ m s}^{-1} \\ L_r &= 1 \text{ m} \\ \delta_r &= 10^{-3} \text{ m} \\ \rho_r &= 1.2 \text{ Kg m}^{-3} \\ E_r &= 500 \text{ V cm}^{-1} \\ \sigma_r &= 15\ \mu\text{C cm}^{-3} \\ \nu_r &= 1.5 \times 10^{-5} \text{ m}^2 \text{ s}^{-1} \end{aligned} \tag{31}$$

For these reference values, the adimensional constants ω and γ defined after Eq.(9) are

$$\begin{aligned} \omega &= 0.15 \\ \gamma &= 62.499 \end{aligned} \tag{32}$$

For comparison we mention that in the classical force-free Blasius solution, and for these reference parameters, the $y-$coordinate y^* corresponding to δ^* (that is, the point at which $\frac{u}{u_e} = 0.95$) is

$$y^* = 1.55 \times 10^{-3}\sqrt{x} \tag{33}$$

Stability of a laminar boundary layer cannot be safely guaranteed for local Reynold numbers greater than about 10^3. Therefore requiring

$$R_S = \frac{\widetilde{u}_e \widetilde{y}^*}{\widetilde{\nu}} \simeq 10^3 \tag{34}$$

one obtains, for the reference parameters, $\widetilde{y}^* \simeq 0.15$ mm. Using (31) the conclusion is that, for these parameters, the laminar part of a force-free boundary layer is only of the order of 1 cm, just a tiny portion of a typical wing.

Now we use the scaling solutions (29) and (28) to obtain an estimate of the effects of a longitudinal electric field. For the constant field case (29) from

$$f'(\delta^*) = 1 - e^{-\delta^*\sqrt{k}} = 0.95$$

and

$$y^* = \delta^*\sqrt{c_2}\sqrt{\frac{\nu\omega}{u_e}} = 0.15$$

using (30) one obtains

$$\sigma_0 = 0.957$$

That is, to insure a constant width boundary layer with local Reynolds number $R_S = 10^3$ (at the point where $\frac{u}{u_e} = 0.95$), one needs a charge density $\widetilde{\sigma}_0$ at $y = 0$, or, in physical units

$$\widetilde{\sigma}_0 = \sigma_0\sigma_r = 14.36\ \mu\text{C cm}^{-3}$$

For the variable field case (28) the estimate depends on the separation of the electrodes. Taking $l(x) = 10$, that is an electrode separation of the order of one centimeter, and the references values for all quantities except for the charge density (namely $E_0 = \frac{g(x)}{l(x)} = 1$, $u_e = 1$, etc.) one obtains $c_4 = 0.15$, $g(x) = 10$, $c_3 = g(x)\xi(x) = 258.2$, $\sqrt{c_2} = \xi(x) = 25.8$, and requiring

$$y^* = \delta^* \sqrt{c_2}\sqrt{\frac{\nu\omega}{u_e}} = 0.15$$
$$f'(\delta^*) = 0.95$$

one finally obtains $a = 39887.77$ leading to

$$\sigma_0 = 0.957$$

the same estimate as above. The large value of a that is obtained shows that the WKB expression (28) is a good approximation for physically interesting parameter values. On the other hand the fact that the same charge density estimate is obtained both in the constant-field and the variable-field cases, shows that it is realistic to consider the field as approximately constant throughout the laminar boundary layer width, as long as a variable charge profile is used.

The above estimates were obtained using the reference values for the kinematic variables. For other values we have the following designing formula (in normalized units)

$$\sigma_0 E_0 = 0.957 \frac{u_e^2 \rho_m}{10^{-6} R_S^2 \nu} \tag{35}$$

2.3 *Numerical results*

For the numerical solution of Eq.(17), with σ given by Eq.(19), I use an implicit finite-difference technique ([16] - [18]). Define $F(x,\eta)$ by

$$F(x,\eta) = \frac{\partial f}{\partial \eta} \tag{36}$$

and

$$\begin{aligned} a_1 &= \frac{1}{\beta}\left(\frac{\partial \beta}{\partial \eta} + \xi \frac{\partial \xi}{\partial x} f + \xi^2 \frac{\partial f}{\partial x}\right) \\ a_2 &= -\frac{\gamma}{\beta u_e \rho_m} \xi^2 E_x \sigma_0 \\ a_3 &= -\frac{\xi^2}{\beta} F \\ a_4 &= \frac{\gamma}{u_e \rho_m \beta} \xi^2 E_x \sigma_0 + \frac{\xi^2}{\beta} \frac{\partial u_e}{\partial x} \end{aligned} \tag{37}$$

Then Eq.(17) becomes

$$\frac{\partial^2 F}{\partial \eta^2} + a_1 \frac{\partial F}{\partial \eta} + a_2 F + a_3 \frac{\partial F}{\partial x} + a_4 = 0 \tag{38}$$

The derivatives are replaced by finite-difference quotients with a variable grid spacing concentrated near $\eta = 0$, where F changes more rapidly. Let $k > 1$ be the ratio between two successive grid spacings in the $\eta-$direction.

$$k = \frac{\eta_{i+1} - \eta_i}{\eta_i - \eta_{i-1}}$$

Then

$$\left(\frac{\partial^2 F}{\partial \eta^2}\right)_{i+1,j} = 2\frac{F_{i+1,j+1} + kF_{i+1,j-1} - (1+k)F_{i+1,j}}{\Delta_2}$$
$$\left(\frac{\partial F}{\partial \eta}\right)_{i+1,j} = \frac{F_{i+1,j+1} - k^2 F_{i+1,j-1} - (1-k^2)F_{i+1,j}}{\Delta_1}$$
$$\left(\frac{\partial F}{\partial x}\right)_{i+1,j} = \frac{F_{i+1,j} - F_{i,j}}{\Delta x}$$
$$\Delta_1 = \eta_{j+1} - \eta_j + k^2(\eta_j - \eta_{j-1})$$
$$\Delta_2 = (\eta_{j+1} - \eta_j)^2 + k(\eta_j - \eta_{j-1})^2$$

Substitution in Eq.(38) yields

$$A_j F_{i+1,j+1} + B_j F_{i+1,j} + D_j F_{i+1,j-1} + G_j = 0 \tag{39}$$

with

$$A_j = \frac{2}{\Delta_2} + \frac{a_1}{\Delta_1}$$
$$B_j = \frac{-2(1+k)}{\Delta_2} - \frac{a_1(1-k^2)}{\Delta_1} + a_2 + \frac{a_3}{\Delta x}$$
$$D_j = \frac{2k}{\Delta_2} - \frac{a_1 k^2}{\Delta_1}$$
$$G_j = a_4 - a_3\frac{F_{i,j}}{\Delta x}$$

The boundary conditions at $\eta = 0$ and $\eta \to \infty$ are known

$$f(i,1) = F(i,1) = 0$$
$$F(i,N) = 1$$

where N is the largest label of the grid, in the $\eta-$coordinate, chosen to be sufficiently large.

Because of the tridiagonal nature of (39) the solution in the line $i+1$ is obtained by the two-sweep method, the recursion relations being

$$F_{i+1,j} = \alpha_j F_{i+1,j+1} + \beta_j$$
$$\alpha_j = -\frac{A_j}{B_j + D_j\alpha_{j-1}}$$
$$\beta_j = -\frac{G_j + D_j\beta_{j-1}}{B_j + D_j\alpha_{j-1}}$$

with $\alpha_1 = 0$ and $\beta_1 = 0$.

To start the integration process there are basically two methods. In the first the integration is performed from left to right in the $x-$coordinate with the grid extended to the left of the airfoil, where the flow is known. With the solution known in the line i, the coefficients A_j to G_j for Eq.(39) are computed at the point (i,j). Notice that $f(i,j)$ is obtained by integration of the solution F.

$$f(i,\eta) = \int_0^\eta F(i,\zeta)d\zeta$$

The integration now proceeds along the lines, from left to right. After a complete pass the process is restarted using now, for the calculation of the coefficients A_j, B_j, D_j and G_j, the old values of F at $(i+1, j)$. The process is repeated several times until the solution stabilizes.

In the second method, which is the one we actually use, the integration process starts from an approximate solution. The scaling solutions derived before are particularly useful for this purpose.

For the calculations we considered an electric field parametrized as in Eq.(18), namely

$$E_x = E_0 \frac{\frac{u_e l^2}{\nu\omega}}{\frac{u_e l^2}{\nu\omega} + \xi^2(x)\eta^2}$$

with $\frac{u_e l^2}{\omega} = 666.66$ which corresponds to $l = 10$, $u_e = 1$ and $\nu\omega = 0.15$. Notice that for these parameters, as pointed out before, the electric field has only a small variation throughout the boundary layer region. For the scaling function we take $\xi(x) = \sqrt{x}$ and consider $\beta = 1$. Then all results depend only on the variable S

$$S = \frac{1}{62.499} \frac{\gamma}{u_e \rho_m} \sigma_0 E_0$$

($S = 1$ when all quantities take the reference values).

In Fig.3 we show a contour plot of the numerical solution for $f'(x, \eta)$ $(= \frac{u}{u_e})$ when $S = 0.6$. From the $x-$dependence of the numerical solutions we may compute the effect of the electric field in extending the laminar part of the boundary layer. By defining, as before, the length of the laminar part as the $x-$coordinate corresponding to a local Reynolds number of 10^3 and denoting by x_0 $(\frac{u}{u_e} = 0.95)$ the force-free value we obtain for the ratio

$$R = \frac{x}{x_0}$$

the results shown in Fig.4. For $S = 0$ we obtain the Blasius solution and as we approach $S = 0.957$, corresponding to the scaling solution, the ratio diverges. The matching of the results in the force-free and the scaling limits is a good control of the numerical algorithm. A general conclusion from the results in Fig.4 is that not much improvement is obtained unless one is able to obtain ionization charge densities of the order of the reference value σ_r.

3 Stochastic ground state simulations

3.1 *Quantum Monte-Carlo and the sign problem*

Numerical simulation is an important technique for many systems of physical importance in Condensed Matter Physics, which cannot be solved exactly. Among

the numerical simulation methods, Quantum Monte-Carlo plays a predominant role. It is used both to compute expectation values of operators and to estimate the ground state wave function. For extensive reviews see [19], [20].

Expectation values. The worldline method

The aim is to compute

$$< A >= \frac{1}{Z}\mathrm{Tr}Ae^{-\beta H} \tag{40}$$

where

$$Z = \mathrm{Tr}e^{-\beta H} = \sum_{i_1} < i_1|e^{-\beta H}|i_1 > \tag{41}$$

with $\beta = \frac{1}{kT}$

For each given Hamiltonian H and observable A, Eq.(40) is the **mathematical model** for the **physical problem**, which is the calculation of the expectation value for A.

To compute the partition function Z, the interval $(0, \beta)$ is divided into L subintervals of length $\Delta\tau = \frac{\beta}{L}$

$$Z = \sum_{i_1,\ldots,i_L} < i_1|e^{-\Delta\tau H}|i_L >< i_1|e^{-\Delta\tau H}|i_{L-1} > \cdots < i_2|e^{-\Delta\tau H}|i_1 > \tag{42}$$

Then, the Hamiltonian is split into diagonalizable pieces

$$H = H_1 + H_2 \tag{43}$$

and Trotter's formula[21]

$$e^{-\Delta\tau H} = e^{-\Delta\tau H_2}e^{-\Delta\tau H_1}\left\{1 + O(\Delta\tau^2)\right\} \tag{44}$$

is used to obtain

$$Z \simeq \sum_{i_1,\ldots,i_{2L}} < i_1|U_1|i_{2L} >< i_{2L}|U_2|i_{2L-1} > \cdots < i_2|U_2|i_1 > \tag{45}$$

where $U = e^{-\Delta\tau H_i}$. The sum (45) is now computed by importance sampling, that is, one generates a sequence of distributions $\{i_1(k)...i_{2L}(k)\}$, $k = 1, 2, 3, ...$ with probabilities

$$p(k) = \frac{< i_1(k)|U_1|i_{2L}(k) > \cdots < i_2(k)|U_2|i_1(k) >}{\sum_{i_1,\ldots,i_{2L}} < i_1|U_1|i_{2L} > \cdots < i_2|U_2|i_1 >} \tag{46}$$

The distributions are, for example, selected by the heat bath algorithm. For each allowed move in the worldlines one computes the ratio R of the new and the old numerators in Eq.(46) and the new configuration is accepted with probability $P = R/(1 + R)$.

The above description applies to the computation of the partition function. To compute expectation values of operators three different cases have to be considered:

(i) Operators that are diagonal in the occupation number representation: We simply average over the intermediate states after statistical equilibrium is attained.

(ii) Non-diagonal operators A conserving particle number within a two-site block: Decompose $A = A_1 + A_2$ and compute $< i_k|A_iU_i|i_{k+1} >$.

(iii) Operators that do not conserve the particle number: The calculation now involves disconnected world lines and is more delicate[22].

In simple cases, for example for a one-dimensional chain of spin-polarized fermions with nearest-neighbor interactions

$$H = -t\sum_i(c_i^\dagger c_i + c_{i+1}^\dagger c_i) + V\sum_i(n_i - \frac{1}{2})(n_{i+1} - \frac{1}{2}) = \sum_{i=1}^{N} H_{i,i+1} \tag{47}$$

($n_i = c_i^\dagger c_i$), choosing the splitting

$$H_1 = \sum_{i \text{ odd}} H_{i,i+1} \quad H_2 = \sum_{i \text{ even}} H_{i,i+1} \tag{48}$$

all terms in Eq.(45) are positive. Therefore the interpretation of the quantities $p(k)$ as probabilities is consistent. Unfortunately for higher dimensional cases the situation is not so favorable.

Projector Monte-Carlo

For the ground state ($T \to 0, \beta \to \infty$) the computation of the partition function is unreliable because $\Delta = \frac{\beta}{L}$ becomes very large. Instead one may use the fact that the quantity

$$e^{-\tau H}|\psi >= e^{-\tau E_0}\left\{|0 >< 0|\psi > + \sum_{n>0} e^{-\tau(E_n - E_0)}\right\} \tag{49}$$

tends to a constant times the ground state when $\tau \to \infty$, provided $< 0|\psi > \neq 0$. Therefore, using Trotter's formula to compute $e^{-\tau H}|\psi >$, we obtain a numerical estimate of the ground state. As initial state $|\psi >$ it is convenient to choose a reasonable approximation of the ground state, for example a variational wave function.

The (minus) sign problem

If not all the terms in Eq.(45) are positive, their interpretation as probabilities, for the importance sampling calculation, is not straightforward. We may still write

$$< A >= \frac{1}{Z}\mathrm{Tr}(AP) = \frac{\mathrm{Tr}(A|P|\mathrm{sign}P)}{\mathrm{Tr}(|P|\mathrm{sign}P)} = \frac{< A\mathrm{sign}P >_{|P|}}{< \mathrm{sign}P >_{|P|}} \tag{50}$$

and use $|P|$ as a probability for the Monte-Carlo simulation. However at low temperature the weight of the positive and negative terms is such that both the numerator and the denominator are small quantities. Therefore, because of the statistical errors, the evaluation of $< A >$ turns out to be quite unreliable.

The physical origin of the sign problem

In the projector Monte-Carlo method, for example, the quantity

$$\psi_\tau = e^{-\tau H}|\phi_{\tau=0} >$$

tends to the ground state when $\tau \to \infty$. The weights that are used for the statistical sampling are proportional to the wave function ψ_τ. However we know that, for fermions, ψ_τ cannot be positive for all configurations. Decompose the wave function into positive and negative parts

$$\psi_\tau = \psi_\tau^+ - \psi_\tau^-$$

At low energies both ψ_τ^+ and ψ_τ^- have non-vanishing projections on the "hard-core" bosonic ground state. Hence, at low temperatures, the simulation spends most of the time below the actual energy of the fermionic ground state (Fig.5). These configurations have to be cancelled in the sum, hence the minus signs. A similar effect is found in frustrated spin systems.

One therefore sees that at low temperatures the Quantum Monte-Carlo method is quite unreliable and a need is felt for alternative simulation methods or ways to circumvent the sign problem. It is also clear that faster computers are not the solution because, more than a problem of statistics, the real issue is that we are trying to obtain a small quantity from the difference of two large quantities.

The sign problem is representation-dependent because, had the exact eigenstates of the Hamiltonian been known, then all Boltzman weights in Eq.(46) would have been positive. It is therefore important to be able to estimate the decay rate of the average sign for each specific representation without having to perform massive Monte-Carlo calculations[23]. For a review of techniques to counter the effects of the minus sign problem, in the context of the auxiliary-field quantum Monte-Carlo method, see Ref.[24]. Here I will discuss an attempt to avoid this problem by the construction of a stochastic process associated to the ground state[25].

3.2 *Ground state processes*

The ground state and the Euclidean dynamics of a quantum system are uniquely associated to a stochastic process. Let ρ and G be the unique invariant measure density and the generator of the stochastic process, such that

$$-\hbar\rho^{\frac{1}{2}}G\rho^{-\frac{1}{2}} = H \tag{51}$$

is the Hamiltonian operator of the physical system. Then $\rho^{\frac{1}{2}}$ is the (zero energy) ground state , i. e. $H\rho^{\frac{1}{2}} = 0$ [26] [27]. From the representation $T = exp\{-tG\}$ for the operator semigroup

$$(T_t f)(x) = \int P(t, x, dy) f(y) \tag{52}$$

associated to the transition functions of the Markov process, it follows that the Euclidean correlations of the theory (Schwinger functions) are obtained from the stochastic correlations of the process. The stochastic process may therefore be used to study the dynamical properties of the quantum system.

Existence of a process with unique invariant measure insures the existence of a Schrödinger picture for the quantum system and stochastic techniques like the theory of small random perturbations of Wentzell and Freidlin[28] [29], may be used to obtain rigorous non-perturbative results, in general not accessible through functional analytic methods[30] [31]. The stochastic process involved in this construction is a classical stochastic process which happens to have $\rho = |\phi|^2$ as its invariant measure and generates a set of Euclidean correlations. It should not be confused with the *quantum probability*[32] [33] process associated to real-time quantum evolutions.

This stochastic formulation of quantum systems has, in the past, been used mostly for boson systems. Here I will concentrate on methods which are also appropriate for fermion systems. In the occupation number representation, fermion systems appear associated to jump processes which, for boson systems, are characteristic of non-local interactions. Because of the formal similarities I will first review the construction of the stochastic process for non-local interactions[27] [34].

Let H be an Hamiltonian operator and $\phi(x)$ its (real) lowest energy eigenstate (which is adjusted to zero energy by the addition of a constant to H).

$$(H\phi)(x) = -\frac{\hbar^2}{2m}\Delta\phi(x) + U(x)\phi(x) + \int d^n y V(x, y)\phi(y) = 0 \tag{53}$$

with $V(x, y)$ real finite and $V(x, y) = V(y, x)$.

From (51) one obtains for the generator of the ground state process with invariant measure $\rho = \phi^2$,

$$(Gf)(x) = \left(\frac{\hbar}{m}\frac{\nabla\phi}{\phi}\nabla\right) f + \frac{\hbar}{2m}\Delta f - \frac{1}{\hbar}\int d^n y \phi^{-1}(x) V(x, y)\phi(y)\{f(y) - f(x)\} \tag{54}$$

Consider now the general differential Chapman-Kolmogorov equation

$$\begin{aligned}\partial_t p(xt|zt') = &-\nabla\cdot\left\{b(x,t)p(xt|zt')\right\} + \tfrac{1}{2}\textstyle\sum_{i,j}\frac{\partial^2}{\partial x_i \partial x_j}\left\{a_{ij}(x,t)p(xt|zt')\right\} \\ &+ \textstyle\int d^n y\left\{W(x|y)p(yt|zt') - W(y|x)p(xt|zt')\right\}\end{aligned} \tag{55}$$

for which the generator $Gf = \lim_{\Delta t \downarrow 0} \frac{(T_{\Delta t} f - f)}{\Delta t}$ is

$$(Gf)(x) = b(x,t)\nabla_x f(x) + \frac{1}{2} a_{ij} \frac{\partial^2}{\partial x_i \partial x_j} f(x) + \int d^n y W(y|x) \{f(y) - f(x)\} \quad (56)$$

Comparing Eqs. (54) and (56) one concludes that the ground state process associated to the non-local Hamiltonian in Eq.(50) has drift, diffusion coefficient and jumping kernel, respectively

$$b = \frac{\hbar}{m} \frac{\nabla \phi}{\phi} \quad (57)$$

$$a_{ij} = \frac{\hbar}{m} \delta_{ij} \quad (58)$$

$$W(y|x) = -\frac{1}{\hbar} \phi^{-1}(x) V(x,y) \phi(y) \quad (59)$$

Notice the association, in processes ruled by non-local interactions, of diffusion and jumping. This will also be typical of processes involving bosons and fermions.

Likewise, if the state space of an Hamiltonian is spanned by a discrete set $\{|f>\}$ of basis states one sees, by analogy with (59), that the jumping process with kernel

$$W(f|f') = -\frac{1}{\hbar} Re\{\frac{1}{< f'|\phi >} < f'|H|f >< f|\phi >\} \quad (60)$$

has invariant density $\rho(f) = | < f|\phi > |^2$ as may easily be checked from the stochastic equation

$$\partial_t \rho(f) = \sum_{f'} \left\{ W(f|f')\rho(f) - W(f'|f)\rho(f) \right\} \quad (61)$$

Notice that in this equation the kernel is defined only up to terms const.$\times \delta_{f,f'}$.

In the Eqs. (57-60) the drift and the jumping kernel are both functions of the ground state wave function ϕ which, in general, is not known for a given Hamiltonian. It is however possible to formulate the dynamical problem in such a way that the drift and the jumping kernel are obtained directly without having to solve the equation $H\phi(x) = \lambda_0 \phi(x)$ for the lowest eigenvalue λ_0. Let ϕ_0 be some initial state non-orthogonal to the ground state. Then, in the $t \to \infty$ limit, the solution $\phi_t = \exp\left\{-\frac{1}{\hbar} Ht\right\} \phi_0$ of the Euclidean equation

$$\hbar \partial_t \phi_t(x) = -H\phi_t(x) \quad (62)$$

converges to the ground state. Namely $\exp\{\frac{1}{\hbar}\lambda_0 t\}\phi_t \to \phi$. Defining

$$b_t = \frac{\hbar}{m} \frac{\nabla \phi_t}{\phi_t} \; ; \; W_t(y|x) = -\frac{1}{\hbar} \phi_t^{-1}(x) V(x,y) \phi_t(y) \quad (63)$$

where ϕ_t is a solution of the Euclidean equation (62), one obtains evolution equations for b_t and $W_t(y|x)$

$$\partial_t b_t(x) = \frac{1}{2}\nabla b_t^2(x) + \frac{\hbar}{2m}\nabla\left(\nabla\cdot b_t(x)\right) - \frac{1}{m}\nabla U(x) + \frac{\hbar}{m}\nabla_x \int d^n y W_t(y|x) \quad (64)$$

$$\begin{gathered}\partial_t W_t(y|x) = W_t(y|x)\int d^n z\left\{W_t(z|y) - W_t(z|x)\right\} + \\ \tfrac{1}{2}W_t(y|x)\left\{2U(x) - \tfrac{m}{\hbar}b_t^2(x) - \nabla\cdot b_t(x) - 2U(y) + \tfrac{m}{\hbar}b_t^2(y) + \nabla\cdot b_t(y)\right\}\end{gathered} \quad (65)$$

As ϕ_t converges to the ground state, the solutions of Eqs.(64) and (65) converge to the drift and jumping kernels associated to the ground state process. Therefore these quantities may be obtained directly by iteration of these equations, without having to solve the Schrödinger equation. In the same way, for the pure jump case associated to the discrete basis $\{|f>\}$ representation, one obtains the (Euclidean) evolution equation

$$\partial_t K_t(f|f') = \sum_g K_t(f|f')\left\{K_t(g|f) - K_t(g|f')\right\} \quad (66)$$

for the quantity

$$K_t(f|f') = -\frac{1}{\hbar}\frac{1}{<f'|\phi_t>} <f'|H|f><f|\phi_t> \quad (67)$$

the jumping kernel being

$$W_t(f|f') = \mathrm{Re}K_t(f|f')$$

Starting from an initial W_0, consistent with (67), this equation converges in the $t\to\infty$ limit to the jumping kernel of the ground state process. We may start for example from

$$W_0(f|f') = -\frac{1}{\hbar}\mathrm{Re} <f|H|f'>$$

which corresponds to consider the initial ϕ_t to have the same projection on all basis states. Although Eq.(66) by itself contains no explicit dynamical information on the interaction potential, it is able to generate the ground state kernel because it preserves under time evolution all the information introduced by the initial condition, namely $\partial_t\{W_t(f'|f)W_t(f|f')\} = 0$. That is, what Eq.(66) does is simply to balance the transition probabilities to make them consistent with an invariant measure while, at the same time, preserving the information about the Hamiltonian that is supplied by the initial condition.

For a many-body quantum system, because the number of possible states is very large, the Euclidean equations (66) are a very large system. If the interactions are short-range, a possible solution is to make the iteration on a small

cluster simulation of the system to obtain an approximation of the ground state stochastic kernels. In favorable cases[35] a good analytical approximation may also be obtained from Eq.(66) or from Feynman integral approximations to the ground state[36]. These kernels may then be used to obtain the correlation functions by simulating the process in a much larger lattice. The simulation uses Eqs. (55) and (61) or the corresponding stochastic differential equations.

Expectation values of observables for the process X_t defined by the stochastic equations are the quantum expectation values in the ground state

$$E\{A(X_t)\} = \int A(x)\rho(x)dx = \int \phi(x)A(x)\phi(x)dx = (\phi, A\phi) \tag{68}$$

The multi-time correlations of the stochastic process are the Euclidean correlations of the quantum system. For example

$$\begin{aligned} E\{A(X_t)A(X_0)\} &= \int (T_t A)(x)A(x)\rho(x)dx = \int \phi^2(x)A(x)\left(e^{-\frac{t}{\hbar}(\phi^{-1}H\phi)}A\right)(x)dx \\ &= \left(\phi, Ae^{-\frac{t}{\hbar}H}A\phi\right) \end{aligned} \tag{69}$$

and similarly for the general n-point correlation function.

Assume that one chooses to compute the stochastic kernels in a basis $\{|f>\}$ that diagonalizes the observable A for which one wishes to obtain the correlation functions. Then, the calculation above also shows that, in such an "adapted occupation number basis", there is no essential difference in the procedures to be used for boson or for fermion systems. In particular the fermion "sign problem" is avoided in this algorithm. The price one pays is to have to solve the auxiliary Euclidean equation (66).

In Ref.[25] this method was applied to the Hubbard model and in Ref.[35] to a modified pairing model. Figs. 6 and 7 show some results for the local antiferromagnetic correlations n_a and pairing n_p as a function of hole doping x_h, in the Hubbard model ($U = 1, t = 1$). The general conclusion is that local antiferromagnetic correlations are not destroyed by hole doping and that the Hubbard model, by itself, is not able to explain hole pairing.

4 Handling non-Gaussian data. Characteristic function and large deviations

4.1 *Non-Gaussian data and the neural computation of the characteristic function*

The aim of principal component analysis (PCA) is to extract the eigenvectors of the correlation matrix, from the data. There are standard neural network algorithms for this purpose[37] [38] [39] [40]. However if the process is non-Gaussian,

PCA algorithms or their higher-order generalizations provide only incomplete or misleading information on the statistical properties of the data.

Let x_i denote the output of node i in a neural network. Hebbian learning[41] is a type of unsupervised learning where the neural network connection strengths W_{ij} are reinforced whenever the products $x_j x_i$ are large. The simplest form is

$$\Delta W_{ij} = \eta x_i x_j \tag{70}$$

Hebbian learning extracts the eigenvectors of the correlation matrix Q

$$Q_{ij} =< x_i x_j > \tag{71}$$

but, if the learning law is local as in Eq.(70), all the lines of the connection matrix W_{ij} converge to the eigenvector with the largest eigenvalue of the correlation matrix. To obtain other eigenvector directions requires non-local laws. These principal component analysis (PCA) algorithms find the characteristic directions of the correlation matrix $(Q)_{ij} =< x_i x_j >$. If the data has zero mean ($< x_i >= 0$) they are the orthogonal directions along which the data has maximum variance. If the data is Gaussian in each channel, it is distributed as a hyperellipsoid and the correlation matrix Q already contains all the information about statistical properties. This is because higher order moments of the data may be obtained from the second order moments. However, if the data is non-Gaussian, the PCA analysis is not complete and higher order correlations are needed to characterize the statistical properties. This led some authors[42] [43] to propose networks with higher order neurons to obtain the higher order statistical correlations of the data. An higher order neuron is one that is capable of accepting, in each of its input lines, data from two or more channels at once. There is then a set of adjustable strengths $W_{ij_1}, W_{ij_1 j_2}, ..., W_{ij_1 ... j_n}$, n being the order of the neuron. Networks with higher order neurons have interesting applications, for example in fitting data to a high-dimensional hypersurface. However there is a basic weakness in the characterization of the statistical properties of non-Gaussian data by higher order moments. Existence of the moments of a distribution function depends on the behavior of this function at infinity and it frequently happens that a distribution has moments up to a certain order, but no higher ones. A well-behaved probability distribution might even have no moments of order higher than one (the mean). In addition a sequence of moments does not necessarily determine a probability distribution function uniquely[44]. Two different distributions may have the same set of moments. Therefore, for non-Gaussian data, the PCA algorithms or higher order generalizations may lead to misleading results.

As an example consider the two-dimensional signal shown in Fig.8. Fig.9. shows the evolution of the connection strengths W_{11} and W_{12} when this signal is passed through a typical PCA algorithm. Large oscillations appear and finally the algorithm overflows. Smaller learning rates do not introduce qualitative modifications in this evolution. The values may at times appear to stabilize, but

large spikes do occur. The reason is that the seemingly harmless data in Fig.8 is generated by a linear combination of a Gaussian with the following distribution

$$p(x) = k(2 + x^2)^{-\frac{3}{2}} \tag{72}$$

which has first moment, but no moments of higher order.

To be concerned with non-Gaussian processes is not a pure academic exercise, because in many applications adequate tools are needed to analyze such processes. For example, processes without higher order moments, in particular those associated with Lévy statistics, are prominent in complex processes such as relaxation in glassy materials, chaotic phase diffusion in Josephson junctions and turbulent diffusion[45] [46] [47]. Moments of an arbitrary probability distribution may not exist. However, because every bounded and measurable function is integrable with respect to any distribution, the existence of the characteristic function f(α) is always assured[44].

$$f(\alpha) = \int e^{i\alpha.x} dF(x) =< e^{i\alpha.x} > \tag{73}$$

α and x are N-dimensional vectors, x is the data vector and $F(x)$ its distribution function. The characteristic function is a compact and complete characterization of the probability distribution of the signal. If, in addition, one wishes to describe the time correlations of the stochastic process $x(t)$, the corresponding quantity is the characteristic functional[48]

$$F(\xi) = \int e^{i(x,\xi)} d\mu(x) \tag{74}$$

where $\xi(t)$ is a smooth function and the scalar product is

$$(x, \xi) = \int dt x(t)\xi(t) \tag{75}$$

$\mu(x)$ being the probability measure over the sample paths of the process.

In the following I will describe an algorithm to compute the characteristic function from the data, by a learning process[49]. The main idea is that, in the end of the learning process, one has a neural network which is a representation of the characteristic function. This network is then available to provide all the required information on the probability distribution of the data being analyzed.

Suppose we want to learn the characteristic function $f(\alpha)$ of a one-dimensional signal $x(t)$ in a domain $\alpha \in [\alpha_0, \alpha_N]$. The α-domain is divided in N intervals by a sequence of values $\alpha_0, \alpha_1, \alpha_2, ..., \alpha_N$ and a network is constructed with $N + 1$ intermediate layer nodes and an output node (Fig. 10).

The learning parameters in the network are the connection strengths W_{0i} and the node parameters θ_i. The existence of the node parameter means that the output of a node in the intermediate layer is $\theta_i \chi_i(\alpha)$, χ_i being a non-linear function.

The use of both connection strengths and node parameters in neural networks makes them equivalent to a wide range of other connectionist systems[50] and improves their performance in standard applications[40]. The learning laws for the network in Fig. 10 are:

$$\begin{aligned} \theta_i(t+1) &= \theta_i(t) + \gamma(\cos\alpha_i x(t) - \theta_i(t)) \\ W_{0i}(t+1) &= W_{0i}(t) + \eta \textstyle\sum_j \left(\theta_j(t) - \sum_k W_{0k}(t)\chi_k(\alpha_j)\theta_k(t)\right)\theta_i(t)\chi_i(\alpha_j) \end{aligned} \tag{76}$$

$\gamma, \eta > 0$. The intermediate layer nodes are equipped with a radial basis function

$$\chi_i(\alpha) = \frac{e^{-(\alpha-\alpha_i)^2/2\sigma_i^2}}{\sum_{k=0}^{N} e^{-(\alpha-\alpha_k)^2/2\sigma_k^2}} \tag{77}$$

where in general one uses $\sigma_i = \sigma$ for all i. The output is a simple additive node. The learning constant γ should be sufficiently small to insure that the learning time is much smaller than the characteristic times of the data $x(t)$. If this condition is satisfied each node parameter θ_i tends to $< \cos\alpha_i x >$, the real part of the characteristic function $f(\alpha)$ for $\alpha = \alpha_i$. The W_{oi} learning law was chosen to minimize the error function

$$f(W) = \frac{1}{2}\sum_j \left(\theta_j - \sum_k W_{0k}(t)\chi_k(\alpha_j)\theta_k\right)^2 \tag{78}$$

One sees that the learning scheme is an hybrid one, in the sense that the node parameter θ_i learns, in an unsupervised way, (the real part of) the characteristic function $f(\alpha_i)$ and then, by a supervised learning scheme, the W_{0i}'s are adjusted to reproduce the θ_i value in the output whenever the input is α_i. Through the learning law (76) each node parameter θ_i converges to $< \cos\alpha_i x >$ and the interpolating nature of the radial basis functions guarantees that, after training, the network will approximate the real part of the characteristic function for any α in the domain $[\alpha_0, \alpha_N]$. A similar network is constructed for the imaginary part of the characteristic function, where now

$$\theta_i'(t+1) = \theta_i'(t) + \gamma(\sin\alpha_i x(t) - \theta_i'(t)) \tag{79}$$

For higher dimensional data the scheme is similar. The number of required nodes is N^d for a d-dimensional data vector $\vec{x}(t)$. For example for the 2-dimensional data of Fig.8 a set of N^2 nodes was used (Fig.11). Each node in the square lattice has two inputs for the two components α_1 and α_2 of the vector argument of $f(\vec{\alpha})$. The learning laws are, as before

$$\begin{aligned} \theta_{(ij)}(t+1) &= \theta_{(ij)}(t) + \gamma(\cos\vec{\alpha}_{(ij)}\cdot\vec{x}(t) - \theta_{(ij)}(t)) \\ W_{0(ij)}(t+1) &= W_{0(ij)}(t) + \\ +\eta \textstyle\sum_{(kl)} &\left(\theta_{(kl)}(t) - \sum_{(mn)} W_{0(mn)}(t)\chi_{(mn)}(\vec{\alpha}_{(kl)})\theta_{(mn)}(t)\right)\theta_{(ij)}(t)\chi_{(ij)}(\vec{\alpha}_{(kl)}) \end{aligned} \tag{80}$$

The pair (ij) denotes the position of the node in the square lattice and the radial basis function is

$$\chi_{(ij)}(\vec{\alpha}) = \frac{e^{-|\vec{\alpha}-\vec{\alpha}_{(ij)}|^2/2\sigma^2_{(ij)}}}{\sum_{(kl)} e^{-|\vec{\alpha}-\vec{\alpha}_{(kl)}|^2/2\sigma^2_{(kl)}}} \tag{81}$$

Two networks are used, one for the real part of the characteristic function, and another for the imaginary part with, in Eqs.(80), $\cos \vec{\alpha}_{(ij)}.\vec{x}(t)$ replaced by $\sin \vec{\alpha}_{(ij)}.\vec{x}(t)$. Figs.12a-b show the values computed by the algorithm for the real and imaginary parts of the characteristic function corresponding to the two-dimensional signal in Fig.8. On the left is a plot of the exact characteristic function and on the right the values learned by the network. In this case we show only the mesh corresponding to the θ_i values. One obtains a 2.0% accuracy for the real part and 4.5% accuracy for the imaginary part. The convergence of the learning process is fast and the approximation is reasonably good. Notice in particular the slope discontinuity at the origin which reveals the non-existence of a second moment.

For a second example the data was generated by a Weierstrass random walk with probability distribution

$$p(x) = \frac{1}{6}\sum_{j=0}^{\infty}\left(\frac{2}{3}\right)^j\left(\delta_{x,bj} + \delta_{x,-bj}\right) \tag{82}$$

and b=1.31, which is a process of the Lévy flight type. The characteristic function, obtained by the network, is shown in Fig. 13. Taking the log(− log) of the network output one obtains the scaling exponent 1.49 near $\alpha = 0$, close to the expected fractal dimension of the random walk path (1.5).

These examples test the algorithm as a process identifier, in the sense that, after the learning process, the network is a dynamical representation of the characteristic function and may be used to perform all kinds of analysis of the statistics of the data.

4.2 *Large deviation analysis of experimental data*

Whenever large fluctuations are observed in a process, there is a feature of the data which has to be carefully considered. This is the deviation from the average values of the process that arise from the finite size of the samples. Trained to think in terms of Gaussian or Poisson processes, some physicists live under the impression that the deviation from asymptotic (average) values, coming from the finiteness of the samples, is a minor well-controlled effect. Think however, for a moment, about a data set $\{x_n\}$ where each point x_n is generated by the multiplication of 1000 numbers, each one being either zero or four with probability $\frac{1}{2}$. The expectation value (or asymptotic average $\lim_{n\to\infty} \xi_n = \frac{1}{n}\sum_{i=1}^{n} x_i$) of the data is 2^{1000}, a huge number. However if an experimental approach is taken, without

prior knowledge of the nature of the process, the result might be quite different. Generating for example the "data" on a computer and analyzing it according to well-established experimental standards, even with statistics unmatched by most current physics experiments, I bet that the result would be zero (and with negligible error bars).

Incidentally, if this multiplicatively generated data set seems far too removed from any physically relevant problem, just remember that random multiplicative processes are thought to be good models for turbulent events, fluctuations in multiparticle production, etc.

Large deviation theory is the set of results that studies deviations from the asymptotic averages when the sample is finite (as it always is in experimental physics). If we are interested in measuring an observable, it is the expectation value or asymptotic average that we want to obtain. Large deviations from this value are, in this sense, exceptional events. What large deviation theory tells us is *how probable exceptional events are.*

4.2.1 *A summary of large deviation theory*

Large deviation theory is the study of fluctuations away from the limit values given by the law of large numbers, that may be observed in empirical statistics, due to the finiteness of the sample.

In probability theory one has a space of elementary events Ω, a probability measure P and a set F of measurable subsets of events. Consider now a sequence (Ω_n, F_n, P_n) of probability spaces and a sequence of random vectors (vector-valued random functions) $\xi_n(\omega)$ $\{\omega \in \Omega_n\}$ with values in a metric space Y.

Assume that ξ_n tends to a limit point y_0 when $n \to \infty$ (law of large numbers). Define the following sequence of probability measures Q_n on the Borel sets of Y

$$Q_n\{A\} = P_n\{\omega : \xi_n(\omega) \in A\} \; ; \; A \in B(Y) \tag{83}$$

The existence of the limit $\xi_n \to y_0$ implies that $Q_n\{A\} \to 0$ if y_0 is not in the closure of A.

The sequence Q_n of probability measures is said to have a *large deviation property* (LDP) if there is a sequence $a_n \to \infty$ $a_n \in N$ and a function $I(y) : Y \to [0, \infty]$ such that:

(a) $I(y)$ is lower semicontinuous (i. e. $y_n \to y \Longrightarrow \liminf I(y_n) \geq I(y)$)

(b) $I(y)$ has compact level sets (i. e. $\{y \in Y : I(y) \leq L\}$ compact $\forall L \in R$)

(c) $\lim_{n\to\infty} \sup \frac{1}{a_n} \log Q_n\{F\} \leq -I(F)$, F closed. $\lim_{n\to\infty} \inf \frac{1}{a_n} \log Q_n\{G\} \geq I(G)$, G open; where $I(F) = \inf I(y)$, $y \in F$.

These two limits are usually represented by the symbol $\asymp$, namely

$$Q_n(dy) \asymp \exp\left(-a_n I(y)\right) dy \tag{84}$$

Furthermore, if $I(\text{int}A) = I(\text{cl}A)$ then

$$\lim_{n\to\infty} \frac{1}{a_n} \log Q_n\{A\} = -I(A) \tag{85}$$

I now list a few large deviation theorems that are useful in the applications

Theorem 1 If the function I exists for a sequence $\{a_n\}$ then it is unique.

Theorem 2 If $Q_n(dy) \asymp \exp(-a_n I(y))dy$ on Y then $I(Y) = 0$.

Theorem 3 (Obtaining the deviation function from the free energy (or pressure))

Let W_n be random vectors with scaling constants a_n. If the following limit exists

$$c(t) = \lim_{n\to\infty} \log E^{P_n}\{\exp(t, Wn)\},\ t \in Y \tag{86}$$

and if $c(t)$ is differentiable, then the deviation function in

$$Q_n(dx) = P_n\{\frac{1}{a_n} W_n \in dx\} \asymp \exp\left(-a_n I(x)\right) dx$$

is the Legendre transform

$$I(x) = \sup_{t\in Y}\{tx - c(t)\}$$

E^{P_n} denotes the expectation value in the probability space (Ω_n, F_n, P_n) and $\frac{1}{a_n} W_n$ plays the role of the random vector ξ_n in the large deviation definition. The method proposed here for the large deviation analysis of the data is based on this result.

Theorem 4 (Laplace-Varadhan method) If $Q_n(dy) \asymp \exp(-a_n I(y))dy$ on Y and f is a function $f : Y \to R$, then

$$\lim_{n\to\infty} \frac{1}{a_n} \log \int \exp\{a_n f(y)\} Q_n(dy) = \sup_{y\in Y}\{f(y) - I(y)\}$$

Theorem 5 (Contraction principle) Let ψ be a continuous mapping $\psi : Y \to W$. If $Q_n(dy) \asymp \exp(-a_n I(y))dy$ on Y then

$$Q_n \circ \psi^{-1}(dz) \asymp \exp(-a_n J(z))dz$$

on W, with

$$J(z) = \inf\{I(y) : y \in Y, \psi(y) = z\}$$

and if z is not in the range of ψ then $J(z) = \infty$.

Theorem 6 Let $Q_n(dy) \asymp exp(-a_n I(y))dy$ on Y , $f : Y$ a continuous bounded function and

$$Q_{n,F}\{A\} = \int_A \frac{\exp(a_n f(y))}{\int_Y \exp(a_n f(z)) Q_n(dz)} Q_n(dy)$$

Then

$$Q_{n,F}\{dy\} \asymp exp(-a_n I_f(y))dy$$

with

$$I_f(y) = I(y) - f(y) - \inf_{z \in Y}\{I(z) - f(z)\}$$

This theorem is useful, for example, to obtain finite volume Gibbs states, where the exponential factor is $\exp\{-\beta H_n(\omega)\}$ and the function $f(y)$ is obtained isolating the leading a_n term in the finite volume energy $-H_n(\omega) = a_n f(\xi_n(\omega)) + o(a_n)$.

 # Levels of description of statistical systems and large deviation levels

Let a statistical phenomenon be described by a sequence of values of a random variable X

$$\cdots X_{-2} X_{-1} X_0 X_1 X_2 \cdots$$

which takes values in a space Y. Y is called the *state space* and the space of sequences Y^Z is called the *path space*. The statistical properties of the phenomenon may be described at three different levels:

(1) By the expectation values of observables

$$< g(X) >= \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} g(X_i) \tag{87}$$

(2) By probability measures on the state space Y

$$\rho(dy) = P\{X_i \in (y, y + dy)\} \tag{88}$$

(3) By probability measures on path space Y^Z. These may be constructed by defining the probabilities $P\{\Sigma\}$ on cylinder sets

$$\Sigma_k = \{\omega : X_i \in \Delta_i, \ .., X_{i+k} \in \Delta_{i+k}\} \tag{89}$$

The finite sample versions of the expectation values, the probability on state space and the probability on path space are:

(a) The mean partial sum

$$\frac{1}{n} S_n = \frac{1}{n} \sum_{i=1}^{n} g(X_i) \tag{90}$$

(b) The empirical measure

$$L_n(\omega, A) = \frac{1}{n}\sum_{i=0}^{n} \delta_{X_i}(\omega)(A) \tag{91}$$

A being a measurable set in Y.

(c) The empirical process

To define a measure in Y^Z out of a finite sample, the sample is repeated infinitely many times in the future and the past generating an element $X(n,\omega)$ in Y^Z. Then the empirical process is

$$R_n(\omega, \Sigma) = \frac{1}{n}\sum_{i=0}^{n-1} \delta_{\sigma^i X(n,\omega)}\{\Sigma\} \tag{92}$$

where σ is the shift operator and Σ a cylinder set in Y^Z.

The large deviations of mean partial sums are called *level 1 large deviations (L. D.)* . The large deviations of the empirical measure are called *level 2 L. D.* and the large deviations of the empirical process are called *level 3 L. D.*

Examples

Level 1 L. D.

$$Q_n(dx) = P\{n^{-1}S_n \in (x, x+dx)\} \asymp \exp(-nI(x))dx \tag{93}$$

For a Gaussian process

$$\rho(dx) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{x^2}{2\sigma^2})dx, \text{one has } I(x) = \frac{1}{2\sigma^2}x^2 \tag{94}$$

and for

$$\rho = \frac{1}{2}\delta_1 + \frac{1}{2}\delta_{-1} \tag{95}$$

the deviation function is

$$\begin{aligned} I(x) &= \tfrac{1}{2}(1-x)\log(1-x) + \tfrac{1}{2}(1+x)\log(1+x),\ |x| \leq 1 \\ &= \infty,\ |x| > 1 \end{aligned} \tag{96}$$

Level 2 L. D.

$$Q_n(d\nu) = P\{\omega : L_n(\omega, \cdot) \in d\nu\} \asymp \exp\left(-nI^{(2)}(\nu, \rho)\right) d\nu \tag{97}$$

For independent identically distributed (I.I.D.) random vectors

$$\begin{aligned} I^{(2)}(\nu, \rho) &= \int_Y \left(\log \tfrac{d\nu}{d\rho}(x)\right) \nu(dx)\ ,\ \text{if}\ \ \nu \ll \rho \\ &= \infty\ ,\ \text{otherwise} \end{aligned} \tag{98}$$

and if the state space Y is finite $Y = \{1, 2, ..., r\}$

$$I^{(2)}(\nu, \rho) = \sum_{i=1}^{r} \nu_i \log\left(\frac{\nu_i}{\rho_i}\right) \ , \quad \text{if} \ \ \nu \ll \rho$$

$$= \infty \ , \ \text{otherwise} \tag{99}$$

(the symbol << means "absolutely continuous with respect to")

Level 3 L. D.

$$Q_n(dP) = P_\rho\{\omega : R_n(\omega, \cdot) \in dP\} \asymp \exp\left(-nI^{(3)}(P, P_\rho)\right) dP \tag{100}$$

The finite dimensional marginals of a probability measure in Y^Z are defined by restriction to cylinder sets pinned down at a finite number of points.

$$\Sigma_1 = \{\omega \in Y^Z : X_1 = y_1\} \tag{101}$$

$$\Sigma_2 = \{\omega \in Y^Z : X_1 = y_1, X_2 = y_2\} \tag{102}$$

etc. Then for the empirical process the one-dimensional marginal

$$R_n(\omega, \Sigma_1) = L_n(\omega, \{y_1\}) \tag{103}$$

gives the empirical measure, the two-dimensional marginal

$$R_n(\omega, \Sigma_2) = M_n(\omega, \{y_1, y_2\}) \tag{104}$$

the empirical pair measure, etc.

Denote by $\pi_\alpha P$ the $\alpha-$dimensional marginal of P. Then for independent identically distributed (I.I.D.) random vectors

$$I^{(2)}(\pi_\alpha P, \pi_\alpha P_\rho) = \sum_\omega \pi_\alpha P\{\omega\} \log \frac{\pi_\alpha P\{\omega\}}{\pi_\alpha P_\rho\{\omega\}} \tag{105}$$

and

$$I^{(3)}(P, P_\rho) = \lim_{\alpha \to \infty} \frac{1}{\alpha} I^{(2)}(\pi_\alpha P, \pi_\alpha P_\rho) \tag{106}$$

Large deviation theory was formulated in its modern form by Donsker and Varadhan and developed and applied in several domains by many authors. Here I have put together a minimum set of results which might be useful for the large deviation analysis of experimental data and simple theoretical models. For further results and applications refer to the standard references[51] [52] [53].

4.2.2 *Large deviation analysis of experimental data*

There are several ways to analyze a set of data points from the point of view of large deviations. One may for example look in a small domain $(\nu, \nu + d\nu)$ in probability space and find for each sample size n the number of times that the empirical measure falls in that interval. The rate of variation of the logarithm of that number with n gives the deviation function at ν. This should of course be repeated for other small domains in probability space to obtain the deviation function throughout the space of probability measures. This method has the shortcoming that, each time the analysis is performed, only a small amount of the available data is used and the statistics is poor.

A better method[54] is to construct the free energy

$$c(t) = \lim_{n\to\infty} \frac{1}{n} \log E^{P_n}\{\exp(t, W_n)\} \ , \ \ t \in Y \tag{107}$$

and obtain the deviation function from the Legendre transform

$$I(x) = \sup_{t\in Y}\{tx - c(t)\} \tag{108}$$

Of course in (107) the $\lim_{n\to\infty}$ is estimated from the approach to the largest possible available sample.

When a deviation function is either obtained from the data or conjectured in some other way, a consistency check is provided by the fact that the deviation function is the deviation function of its own distribution

$$Q_n(dI) \asymp \exp(-a_n I) dI \tag{109}$$

This follows from the contraction principle (theorem 5) using I as the function ψ. This property is useful, for example, to check whether the relative entropy

$$I_r = \sum_{i=1}^{r} \nu_i \log\left(\frac{\nu_i}{\mu_i}\right) \tag{110}$$

(μ being the empirical measure computed for the largest possible sample) is a good guess for the deviation function. Recall that the relative entropy coincides with the deviation function if the events are independent.

A simple example shows how the large deviation analysis provides insight on the statistical structure of the data. A multiplicative process $\{X_n\}$ was constructed with expectation value 1, i. e.

$$\lim_{n\to\infty} \frac{1}{n} W_n = 1$$

$$W_n = \sum_{i=1}^{n} X_i$$

(by the law of large numbers) and used to generate data sets which are then analyzed for large deviations. This is done by constructing a numerical approximation to the free energy and its Legendre transform. Figs.14 shows a typical sets of 10^4 events (bin width=10^{-6}). The sample average in this set is 0.285, very far from the asymptotic value. From a simple examination of the distribution of the data it would be hard to guess that one is so far away from the real average value of the process.

For the numerical computation of the free energy the data is divided into k slices of length n, such that $k \times n$ equals the total number of data points and one considers

$$c_{k,n}(t) = \frac{1}{n} \log \left(\frac{1}{k} \sum_{j=0}^{k-1} \exp \left\{ t W_j^{(n)} \right\} \right) \tag{111}$$

where

$$W_j^{(n)} = \sum_{s=jn}^{jn+n-1} X_s$$

A numerical estimate of the limit (107) is obtained by increasing n and decreasing k maintaining the product fixed. In practice this should stop at a level where there still is a sufficient number of data slices to have reasonable statistics in the estimation of $E\{\exp(t, W_n)\}$. In this example we stop at $k = n = 100$. Once the free energy is computed the Legendre transform is easily computed from (108).

Figs.15 and 16 show the numerical estimates for the free energy and the deviation function obtained from the data in Fig.14. Recall that the exact deviation function should have a unique minimum at the average value of the process. Instead, the function that is obtained, has a large flat minimum region. This shows that the sample average cannot be guaranteed to be close to the actual average value. It could be anywhere in the flat region or above. The solid conclusion therefore is that there is insufficient data to make any statement about the average value of the process. This conclusion which comes out very clearly from the examination of the deviation function might indeed be very difficult to reach from the data distribution itself. There is nothing magic in this result. It is simply the effect of the exponential sum in (111) which, as soon as t becomes positive, emphasizes large events, whenever they exist, and makes $c(t)$ grow for $t > 0$.

This example concerns level-1 large deviation properties. For level-2 and level-3 the method is similar, choosing for W_n the appropriate random vector associated to the slicing of the data.

5 A method for fast track recognition in high-energy colliders

The new generation of experiments in high energy physics will have to deal with an unrivalled wealth of information, which has to be treated in an extremely small amount of time (e.g. 15 ns for the LHC). This means that data analysis and track recognition will be a quite complex problem in the LHC, both on-line and off-line. Massive parallelism is compulsory and neural-like algorithms have been proposed in this context[55]. Most papers that adopt this type of algorithm tend to follow an approach in which the connections between the units in the neural architecture are adaptable. This approach is only necessary when very little is known about the function relating the data set to the features we want to detect. If we have some prior knowledge about the function, we may eliminate all (or almost all) adaptability in the network and define an architecture that represents the desired input-output mapping. This is, of course, of primordial interest when one seeks to implement the feature detector in hardware. This approach led to a cellular automata spot reduction technique for particle recognition in e.m. calorimeters[56] and Cherenkov detectors (RICH) [57]. In Ref.[58] a simple cellular automata algorithm was developed to recognize a large number of spirals in a given pattern. The algorithm may be implemented as a massively parallel architecture, the connections between the units being non-adaptable. The algorithm works remarkably well, considering its simplicity, and opens the possibility of track recognition even at the on-line level.

 # The method of spiral reconstruction

Most detectors proposed for the LHC use a strong magnetic field applied parallel to the detector axis. This means that the particles will follow spiral trajectories in the detector, with pitch angle and diameter depending on momentum (and, of course, on the magnitude of the applied magnetic field, which is assumed to be uniform). To reconstruct a spiral one needs the coordinates of four points, at most. Less than four points are needed however, if one assumes that a point in the spiral is fixed, that is the interaction point, and that the spiral axis is along the magnetic field. Of course it might happen that a particle travels some time before it decays (a B meson, for instance, may travel a few millimeters) but let us, for the moment, assume that all spirals stem from a single point. Since the spiral axis is known, only two points are needed to determine all the spiral parameters. Thus, one has the following set of equations for the spiral

$$
\begin{aligned}
x &= \tfrac{v_{0\perp}}{\omega}\left(\cos\phi - \cos\left(\omega t + \phi\right)\right) \\
y &= \tfrac{v_{0\perp}}{\omega}\left(\sin\left(\omega t + \phi\right) - \sin\phi\right) \\
z &= v_{0z} t
\end{aligned}
$$

where v_0 is the initial speed of the particle and $\omega = \frac{qB}{E}$ is the cyclotron frequency for a particle of charge q and energy E in a magnetic field of strength B. In terms

of the coordinates of two points on a given spiral (using cylindrical coordinates, see Fig.17) one has the following set of equations

$$\frac{v_{0z}}{\omega} = b = \frac{z-z'}{2(\theta-\theta')}$$
$$\psi = -\phi = \frac{1}{2}\left(-\frac{z+z'}{2b} + \theta + \theta'\right)$$
$$\frac{v_{0\perp}}{\omega} = a = \frac{r}{2\sin(\theta-\psi)}$$
$$a' = \frac{r'}{2\sin(\theta'-\psi)}$$

In the case considered here, all spirals are assumed to have the same origin and therefore, for two points (r, θ, z) and (r', θ', z'), $a = a'$ is a consistency condition. However, if it is assumed that the origin of the spirals has a random position on the beam axis (this will be the case in the LHC, a problem that will have to be dealt with to solve pile-up events[59]) the a and a' equations may be used to determine this position.

For every pair of hits in the detector there is a point in the three-dimensional parameter space (a, b, ψ). From the pairs that belong to the same spiral a cluster of neighboring points is reconstructed in parameter space. Of course there are also combinations of hits taken from different spirals, but the corresponding points in parameter space do not accumulate in a cluster, they rather yield a diffuse cloud. Notice that it is neither necessary nor advisable to take all possible pairs of points in the data set in order to reconstruct the spirals. This would lead to a waste of computer time because of the combinatorics. Only neighboring points have to be considered. Applying the spot-reduction technique described in Ref.[56] to the parameter space (a, b, ψ) one eliminates the isolated points and, at the same time, finds the center of gravity of the clusters. The spot-reduction algorithm has two separate phases: first it makes two or three partial weight transfers and then two or three total weight transfers. In this way the centre of each individual cluster is singled out with good precision.

Fig.18 shows some results with the algorithm applied to a set of spirals. As may be seen, the reconstruction is quite successful considering the simplicity of the algorithm. The time consumption of the algorithm grow like N^2, N being the number of hits in the detector (which may be subdivided into multiple independent sectors). The execution speed in the spot-reduction algorithm is independent of N (if implemented on a parallel machine). Hence the algorithm is in principle much faster than a conventional road-finder type algorithm, where the computer time grows like a factorial. Also, with growing track multiplicity, the probability of failure of such an algorithm increases with a consequence degradation in reconstruction efficiency. However, in the algorithm described here, it suffices to increase the number of cells into which one divides the parameter space. This does not affect the execution time.

6 Adaptive Monte Carlo integration

The need to integrate functions with (integrable) singularities occurs frequently in numerical work. For simple functions and one-dimensional integrals, standard techniques of truncation of the interval, change of variables, subtraction of the singularity, etc., are applicable[60] [61]. However, for multidimensional integrals and when the functions are so complex that their behavior is essentially unknown, it is convenient to use an integration procedure that automatically adapts itself to the rate of variation of the integrand at each point.

An algorithm (RIWIAD)[62] of Monte-Carlo integration with automatic interval adaptation for multidimensional integrals has been developed by several authors[63] [64] [65] and used in many applications. The RIWIAD algorithm is appropriate to handle integrals with singularities lying on the boundary of the integration volume, which is taken to be the unit hypercube. In each iteration the algorithm computes the variance in the slices of the integration volume associated to the subintervals in each integration axis, that is, for the n'th interval in the i'th axis

$$\sigma_{in}^2 = \sum_{n_1 \cdots n_d} \sigma_{n_1 \cdots n_d}^2 \delta_{n_i, n} \tag{112}$$

In the following iteration the interval sizes are adapted (with a fixed number of points per interval) to ensure that regions of higher variance will have a higher density of sampling points. From the very nature of the algorithm it follows that, in general, it handles well integrands with singularities (or fast rates of variation) lying on hyperplanes perpendicular to the integration axis. However if the singularity hyperplanes lie obliquely to the axis (skew singularities) not much improvement is to be expected from interval adaptation.

To construct an algorithm that handles skew singularities, the obvious brute force method would be to keep in store, from iteration to iteration, the information about the subintervals and variances $\sigma_{n_1 \cdots n_d}$ in all subvolumes of the integration domain (instead of just in the hyperslices perpendicular to each axis) and adapt the number of points in each subvolume accordingly. Notice however that, if m is the number of intervals in each axis, the number of subvolumes grows as m^d whereas the number of hyperslices grows only as $m \times d$, with the dimension d. Thus, for high dimensions and a fine interval division, this method is too demanding in memory requirements. This is the reason why, in RIWIAD, the interval adaptation method is preferred.

The program TSAIR (two-step adaptive integration routines)[66] contains two algorithms that handle satisfactorily skew singularities or fast rates of variation in skew hyperplanes and keep the memory requirements at the same level as RIWIAD. All integrations to be executed by the program should be normalized to the unit hypercube.

ALGORITHM 1

In this algorithm one aims at obtaining minimal variance, for a fixed total

number L of sampling points. If one allows the number of points in each subvolume to vary, the problem consists in minimizing the function

$$\sigma^2 = \sum_{n=1}^{N} \frac{V_n^2}{L_n - 1} x_n \tag{113}$$

by variation of L_n with the subsidiary condition

$$\sum_{n=1}^{N} L_n = L \tag{114}$$

L_n is the number of sampling points in the subvolume V_n , N is the total number of subvolumes and

$$x_n = \left\langle f^2 \right\rangle_n - \langle f \rangle_n^2 \tag{115}$$

is the second order moment of the function in the subvolume n. Applying the Lagrange multiplier method, the result is

$$L_n = \frac{V_n \sqrt{x_n}}{\sum_{n=1}^{N} V_n \sqrt{x_n}} (L - N) + 1 \tag{116}$$

What the algorithm does is to start with a certain number of iterations where it proceeds to interval adaptation as in RIWIAD. The number of such iterations is never smaller than a number chosen by the user and supplied as data. After these iterations, if the variance is still decreasing at a rate faster than a certain amount (10%), the algorithm continues, automatically, to make interval adaptations until this limit is reached.

Starting with the last of the fixed RIWIAD iterations the program computes, and keeps in memory, the value $\sum_{n=1}^{N} V_n \sqrt{x_n}$. Once the interval adaptation step is completed, the program switches to a second step where the interval structure is kept fixed and the number of points per subvolume L_n is allowed to vary to approach the optimum value of Eq.(116). However, to apply Eq.(116) one needs information on $V_n \sqrt{x_n}$ for each subvolume and, as discussed above, to keep these values from iteration to iteration would overburden the memory requirements.

The procedure that is used is the following: each time the program starts sampling inside a subvolume, it first computes a small number y of points. Using these points an estimate for L_n is then obtained from Eq.(116). If the estimate for L_n is larger than the number y of points already sampled, the computer samples an additional $L_n - y$ points (up to a maximum of Z fixed by the user). Otherwise it skips to the next subvolume. The algorithm combines interval adaptation and a variable number of points per subvolume with essentially the same memory requirements as RIWIAD.

ALGORITHM 2

The point of view in this algorithm is the computation of the integral for a specified variance with the minimum possible number of points. The mathematical problem to be solved is the minimization of the function

$$L = \sum_{n=1}^{N} L_n$$

with subsidiary condition

$$\sigma^2 = \sum_{n=1}^{N} \frac{V_n^2}{L_n - 1} x_n$$

By variation of L_n and the Lagrange multiplier method the result is

$$L_n = \frac{\sum_{i=1}^{N} V_i \sqrt{x_i}}{\sigma^2} V_n \sqrt{x_n} + 1 \tag{117}$$

In this algorithm a uniform interval division is used. In the first iteration one computes the integral, the variance and also the sum $\sum_{i=1}^{N} V_i \sqrt{x_i}$. On the second and last iteration, by the same method as described for Algorithm 1, one computes an estimate for $V_n \sqrt{x_n}$ using the first few points sampled in each subvolume. From this estimate and Eq.(117) L_n is then computed and the rest of the necessary points sampled in the subvolume n.

For illustration Fig.19 shows the behavior, from iteration to iteration, of algorithm 1 applied to the function

$$f(x,y) = -\log\left\{|x-y| \,.\, |x-1| \,.\, |y-1| \,.\, |2x+y-2| \,.\, \left|\frac{1}{2}x + y - \frac{1}{2}\right|\right\}$$

which possesses both skew singularities and singularities in lines perpendicular to the axis. The upper plot shows the variance normalized for the same number of points per iteration. As expected, in the first few iterations, one obtains some improvement ($\sim$ 6%) with interval adaptation. However, after the third iteration, no further improvement is observed. After the fifth iteration the interval adaptation ceases, the second phase of algorithm 1 becomes active and a new significant improvement in the variance ($\sim$ 24%) is obtained. The lower plot displays the number of sampled points per iteration. After the fifth iteration this number is no longer fixed. However it is seen that it deviates very little from the value used in the interval adaptation step. This is a good check on the reliability of the method used to estimate the quantities $V_n \sqrt{x_n}$.

References

[1] P. Marksteiner; Comp. Phys. Commun. 97 (1996) 16.

[2] H. Schlichting; *Boundary-Layer Theory* 6th. edition, MacGraw Hill, New York 1968.

[3] A. D. Young; *Boundary Layers*, BSP Professional Books, Blackwell, Oxford 1989.

[4] N. M. El-Hady; Phys. Fluids A3 (1991) 393.

[5] R. D. Joslin and P. J. Morris; AIAA Journal 30 (1992) 332.

[6] R. A. Antonia and Y. Zhu; Phys. Fluids 7 (1995) 2465.

[7] A. D. Lucey and P. W. Carpenter; Phys. Fluids 7 (1995) 2355.

[8] D. M. Bushnell and J. N. Hefner (Eds.); *Viscous drag reduction in boundary layers*, Progress in Astronautics and Aeronautics, vol. 123, American Institute of Aeronautics and Astronautics, Washington 1990.

[9] A. K. Gailitis and O. A. Lielausis; Appl. Magnetohydrodynamics, Trudy Inst. Fis. AN Latv. SSR 12 (1961) 143.

[10] A. B. Tsinober and A. G. Shtern; Magnetohydrodynamics 3 (1967) 103.

[11] H. K. Moffat; J. Fluid Mech. 28 (1967) 571.

[12] A. B. Tsinober; *MHD flow drag reduction*, in [8], page 327.

[13] C. Henoch and J. Stace; Phys. Fluids 7 (1995) 1371.

[14] C. B. Reed and P. S. Lykoudis; J. Fluid Mech. 89 (1978) 147.

[15] R. Vilela Mendes and J. A. Dente; *Boundary layer control by electric fields: A feasibility study*, Lisbon preprint IFM 4/96.

[16] F. G. Blottner; AIAA Journal 8 (1970) 193.

[17] R. T. Davis; AIAA Journal 8 (1970) 843.

[18] H. H. Hamilton II, D. R. Millman and R. B. Greendyke; *Finite-difference solution for laminar or turbulent boundary layer flow over axisymmetric bodies with ideal gas, CF_4 or equilibrium air chemistry*, NASA Langley Research Center report no. L-17102, 1992.

[19] W. von der Linden; Phys. Reports 220 (1992) 53.

[20] L. Mitas; Comp. Phys. Commun. 96 (1996) 107.

[21] M. Suzuki; Prog. Theor. Phys. 56 (1976) 1454.

[22] J. E. Hirsch, R. L. Sugar, D. J. Scalapino and R. Blankenbecler; Phys. Rev. B26 (1982) 5033.

[23] N. Hatano and M. Suzuki; Phys. Lett. A163 (1992) 246.

[24] J. H. Samson; Int. J. Mod. Phys. C6 (1995) 427.

[25] S. Eleutério and R. Vilela Mendes; Physical Review B50, 5035 (1994).

[26] S. Albeverio, R. Høegh-Krohn and L. Streit; J. Math Phys. 18 (1977) 907; 21 (1980) 1636.

[27] R. Vilela Mendes; J. Math. Phys. 27 (1986) 178.

[28] A. D. Wentzell and M. I. Freidlin; Russ. Math. Surv. 25 (1970) 1; *Random Perturbations of Dynamical Systems*, Springer, Berlin, 1984.

[29] A. Friedman; *Stochastic Differential Equations and Applications*, vol. 2, Chap. 14, Academic Press, New York, 1976.

[30] G. Jona-Lasinio, F. Martinelli and E. Scoppola; Commun. Math. Phys. 80 (1981) 223; Phys. Rep. 77 (1981) 313.

[31] R. Vilela Mendes; Phys. Rev. D37 (1988) 1608; Phys. Lett. B202 (1988) 369 ; Phys. Lett. B223 (1989) 83.

[32] R. L. Hudson and K. R. Parthasarathy; Commun. Math. Phys. 93 (1984) 301.

[33] P. A. Meyer; *Elements de Probabilités Quantiques* in *Séminaire de Probabilités XX - XXIII*, Lecture Notes in Mathematics 1204, 1247, 1321, 1372, Springer 1986, 1987, 1988, 1989.

[34] R. Vilela Mendes; Phys. Lett. A113 (1985) 187.

[35] R. Vilela Mendes and S. Eleutério; *Pairing and ground state correlations*, in preparation.

[36] R. Vilela Mendes; Z. Phys. C - Particles and Fields, 54 (1992) 273.

[37] T. D. Sanger; Neural Networks 2, 459 (1989).

[38] E. Oja; Int. J. of Neural Systems 1, 61 (1989).

[39] E. Oja; Neural Networks 5, 927 (1992).

[40] J. A. Dente and R. Vilela Mendes; Network: Computation in Neural Systems 7 (1996) 123.

[41] D. O. Hebb; *The organisation of behavior*, Wiley, New York,1949.

[42] W. R. Softky and D. M. Kammen; Neural Networks 4, 337 (1991).

[43] J. G. Taylor and S. Coombes; Neural Networks 6, 423 (1993).

[44] E. Lukacs; *Characteristic functions*, Griffin, London 1970.

[45] M. F. Shlesinger, G. M. Zaslavsky and J. Klafter, Nature 363, 31 (1993).

[46] G. Zumofen and J. Klafter, Phys. Rev. E47, 851 (1993).

[47] G. Zumofen and J. Klafter, Europhys. Lett. 25, 565 (1994).

[48] T. Hida, *Brownian motion*, Springer, Berlin 1980.

[49] J. A. Dente and R. Vilela Mendes; *Characteristic functions and process identification by neural networks*, Lisbon preprint IFM 1/96.

[50] J. Doyne Farmer, Physica D42, 153 (1990).

[51] S. R. S. Varadhan, *Large deviations and applications*, SIAM, Philadelphia, 1984.

[52] D. W. Stroock, *An introduction to the theory of large deviations*, Springer, New York, 1984.

[53] R. S. Ellis, *Entropy, large deviations and statistical mechanics*, Springer, New York, 1985.

[54] J. Seixas and R. Vilela Mendes; Nuclear Physics B383, 622 (1992).

[55] J. R. Hansen; in *New Computing Techniques in Physics Research*, ed. D. Perret-Gallix, World Scientific, Singapore 1992.

[56] T. Altherr and J. Seixas; in *New Computing Techniques in Physics Research*, CNRS 1990.

[57] T. Altherr and J. Seixas; Nucl. Instr. and Meth. A317 (1992) 335.

[58] T. Altherr, R. Vilela Mendes and J. Seixas; Nucl. Instr. and Meth. A332 (1993) 284.

[59] G. Tonelli; in *New Computing Techniques in Physics Research*, ed. D. Perret-Gallix, World Scientific, Singapore 1992.

[60] P. J. Davis and P. Rabinowitz; *Numerical integration*, Blaisdell Publ. 1967.

[61] G. J. Mata and E. Pestana; Comp. Phys. Comm. 92 (1995) 225.

[62] CERN program library D114, RIWIAD (GENLIB).

[63] W. Czyz, G. C. Sheppey and J. D. Walecka; Nuovo Cimento 34 (1964) 420.

[64] A. J. Dufner; *Proc. 1st. Coll. on Advanced Computing Methods in Theor. Phys.*, Marseille 1970.

[65] B. E. Lautrup; *Proc. 2nd. Coll. on Advanced Computing Methods in Theor. Phys.*, Marseille 1971.

[66] J. C. Romão and R. Vilela Mendes; Computer Journal 21, 355 (1978).

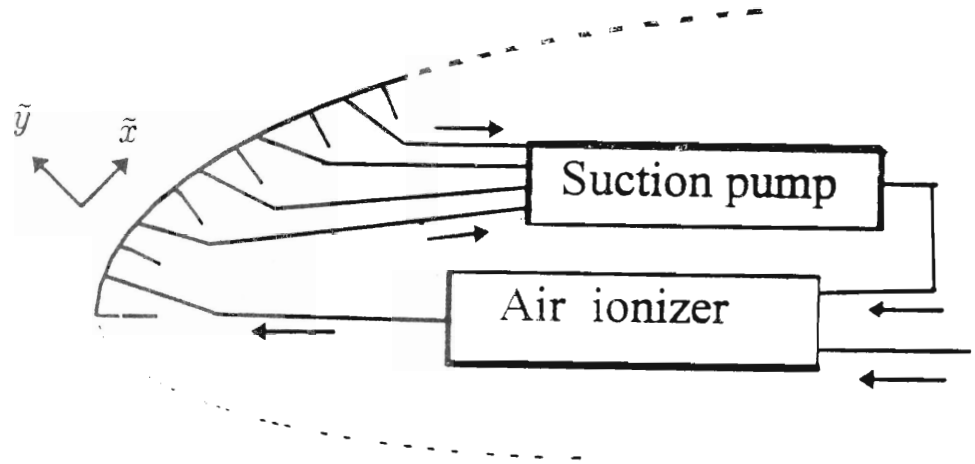


Fig.1 An airfoil transversal cut showing ionized air injection, suction pump and plate electrodes.

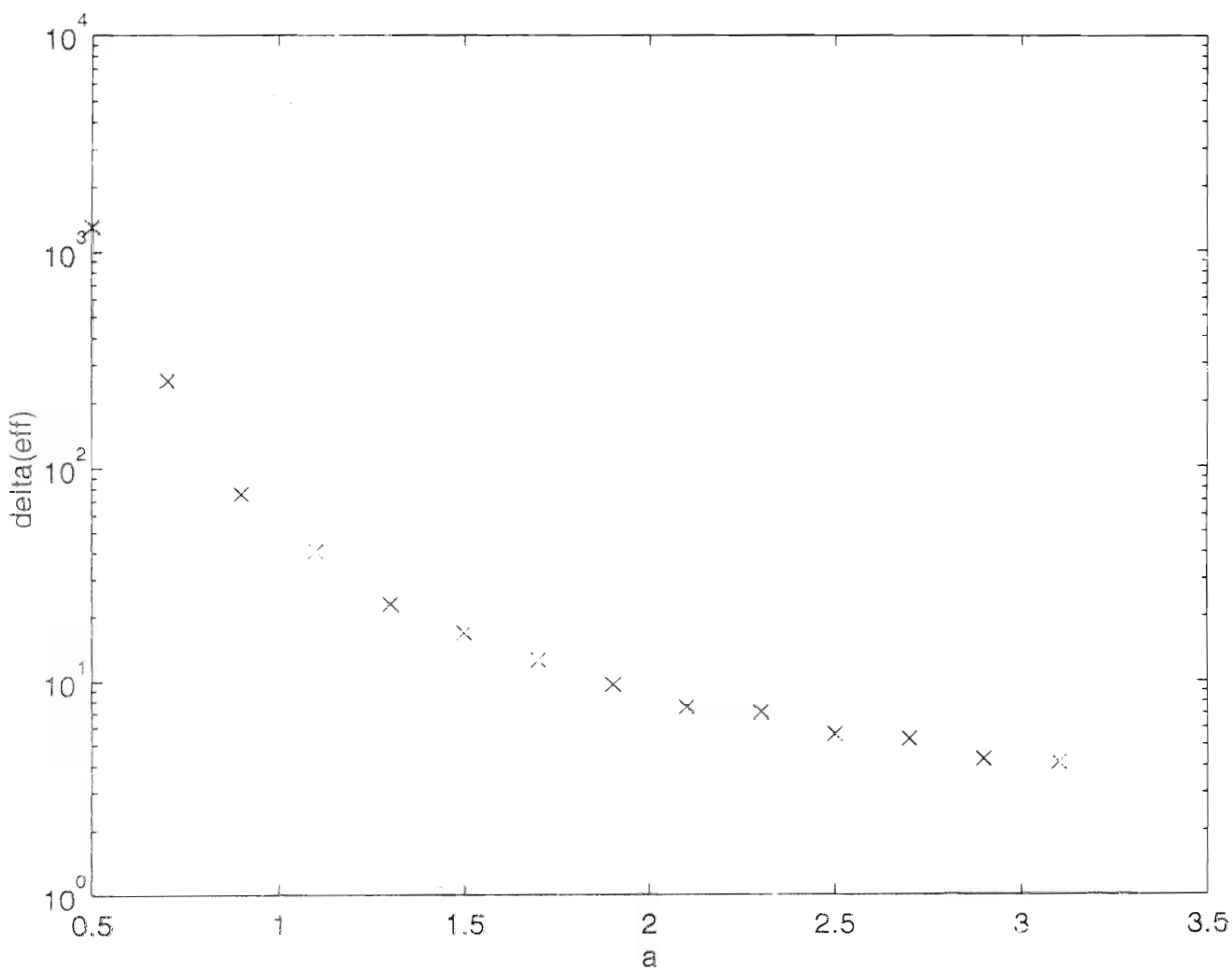


Fig.2 Effective boundary layer width δ^* $\left(f'(\delta^*) = \frac{u}{u_e} = 0.95\right)$ for the constant width scaling solution.

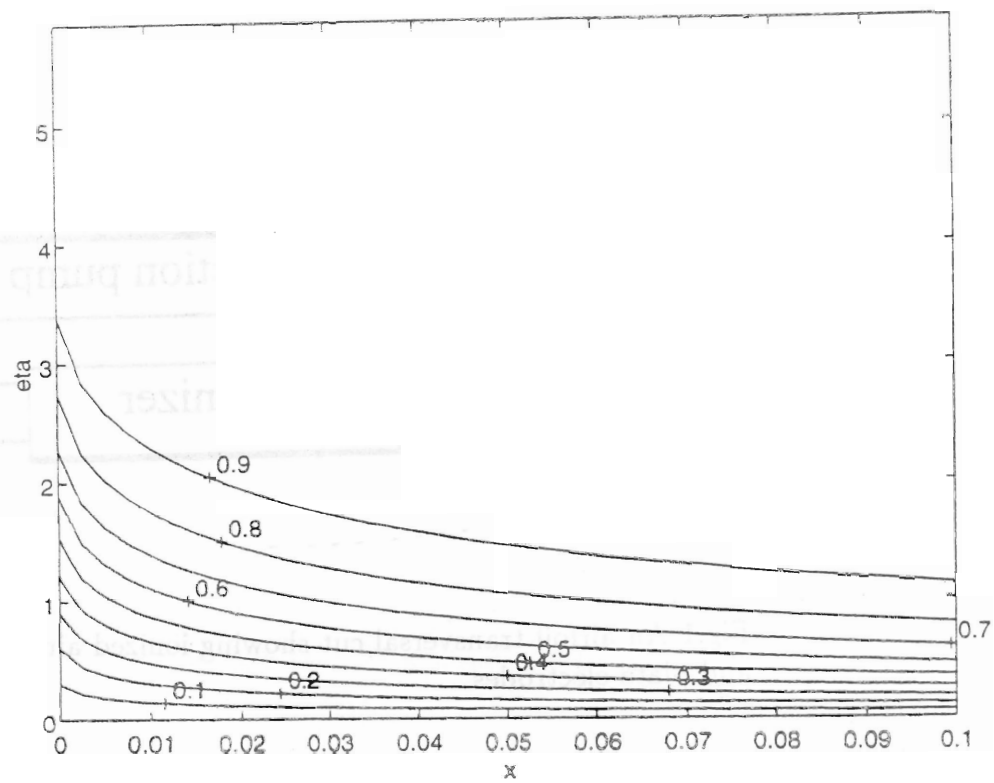


Fig.3 Contour plot of $f'(x,\eta)$ for $S = 0.6$.

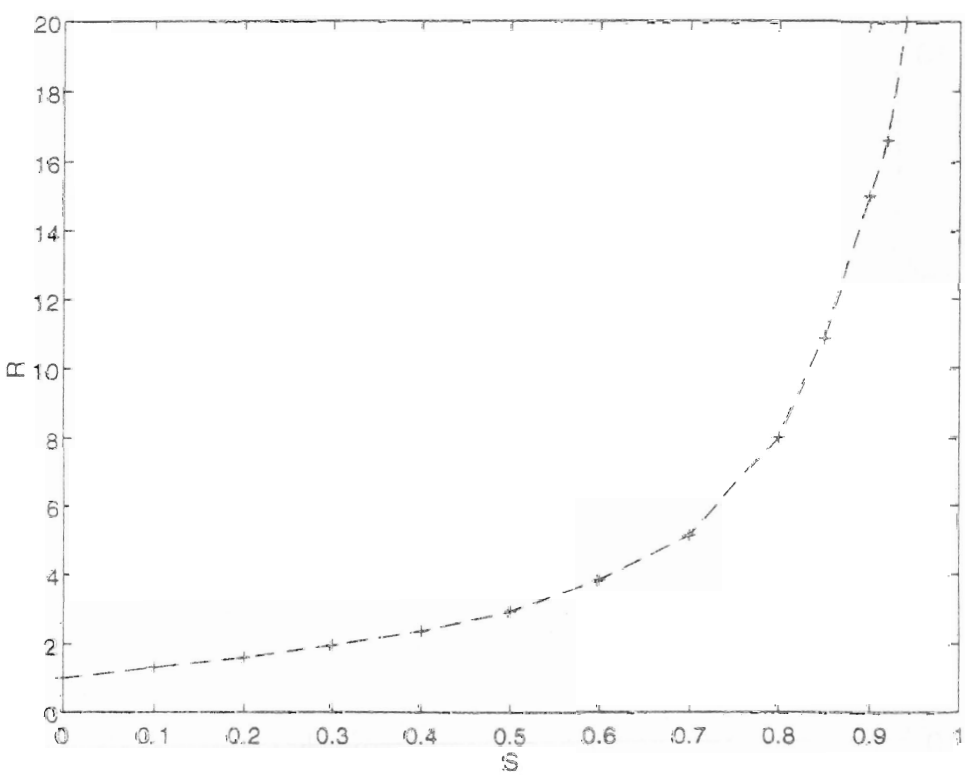


Fig.4 Ratio of boundary layer laminar regions with and without electric field control.

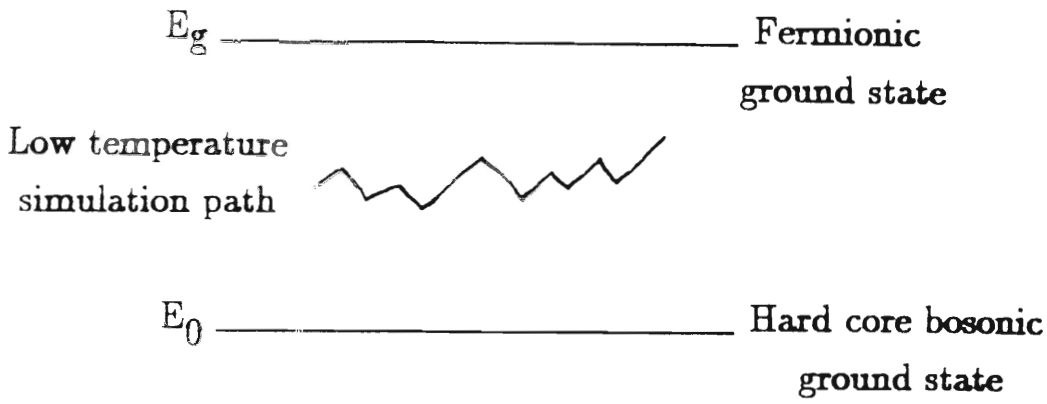


Fig.5 The physical origin of the sign problem

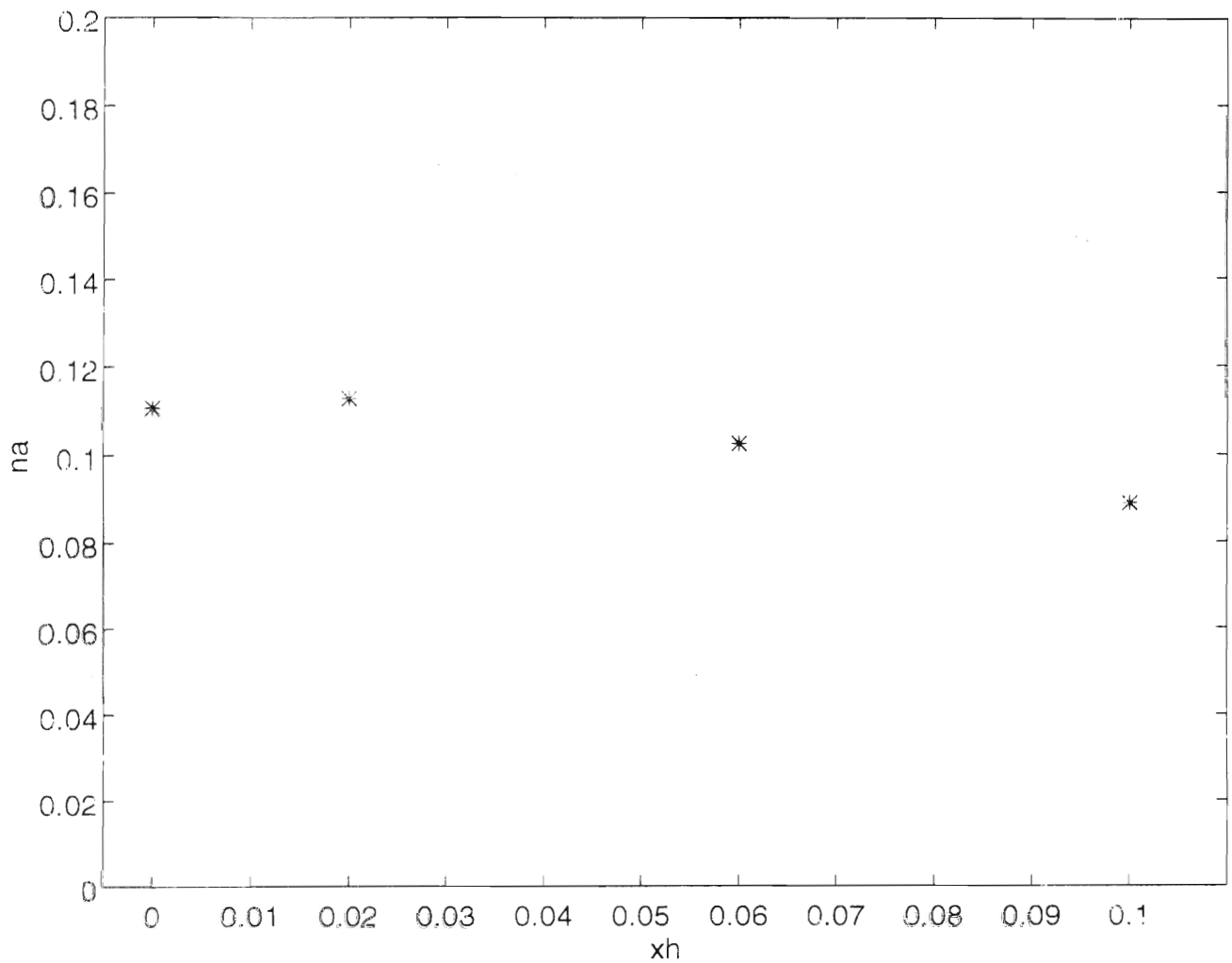


Fig.6 Local antiferromagnetic correlations in the Hubbard model

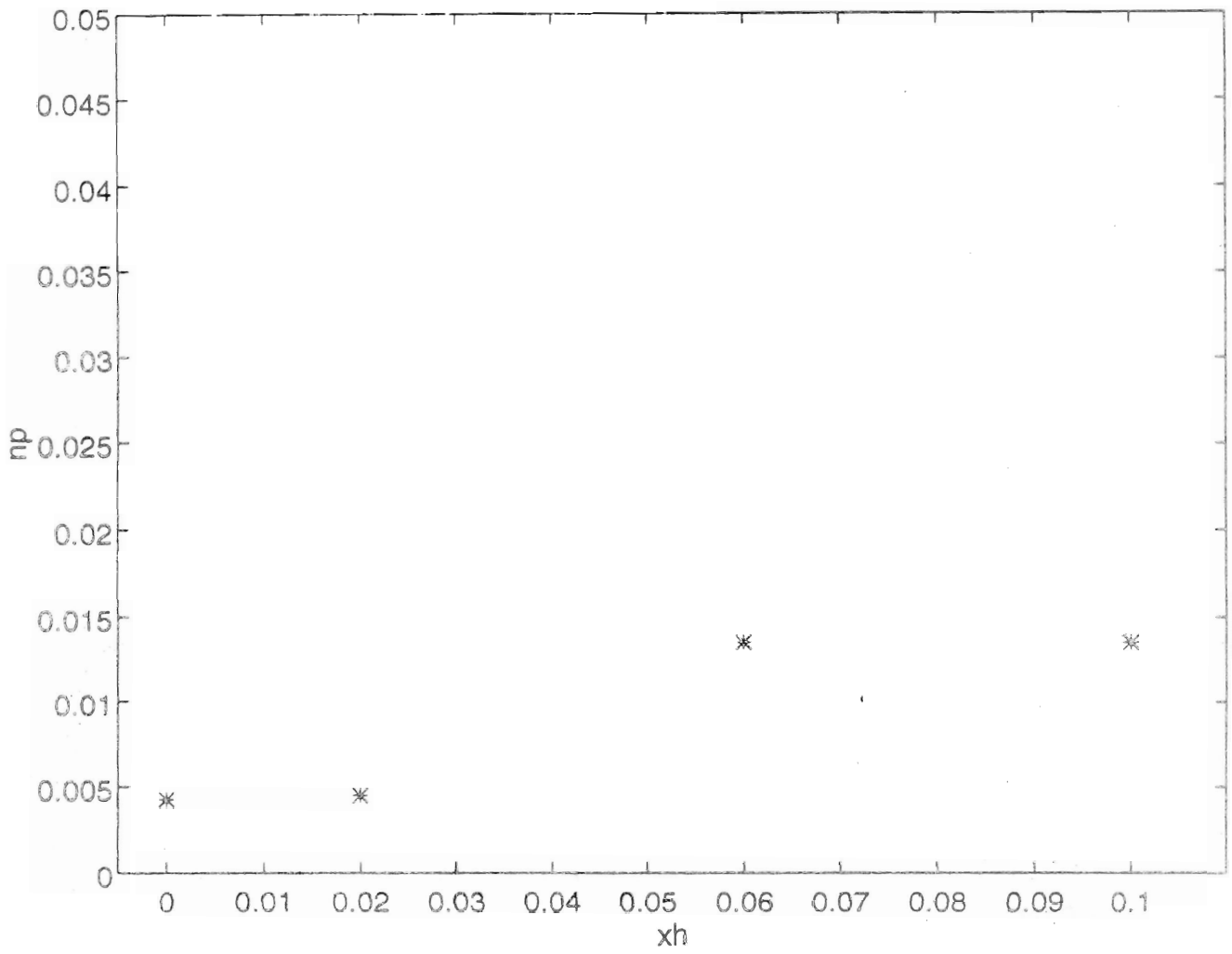


Fig.7 Pairing correlations in the Hubbard model

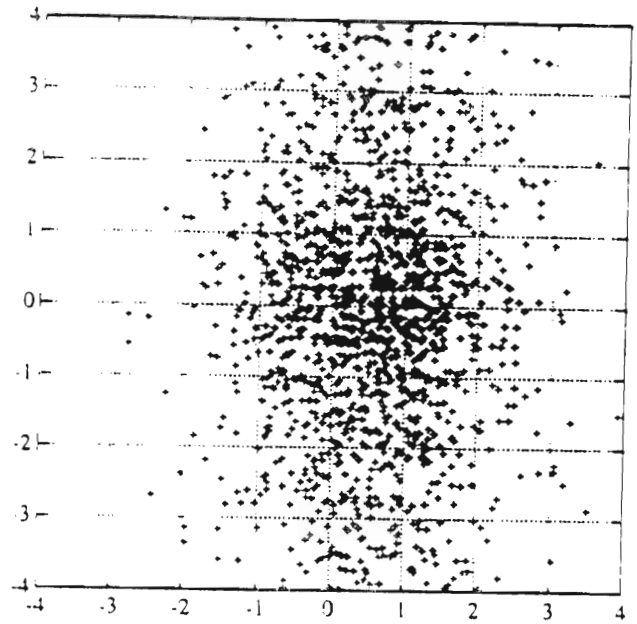


Fig.8 A two-dimensional test signal

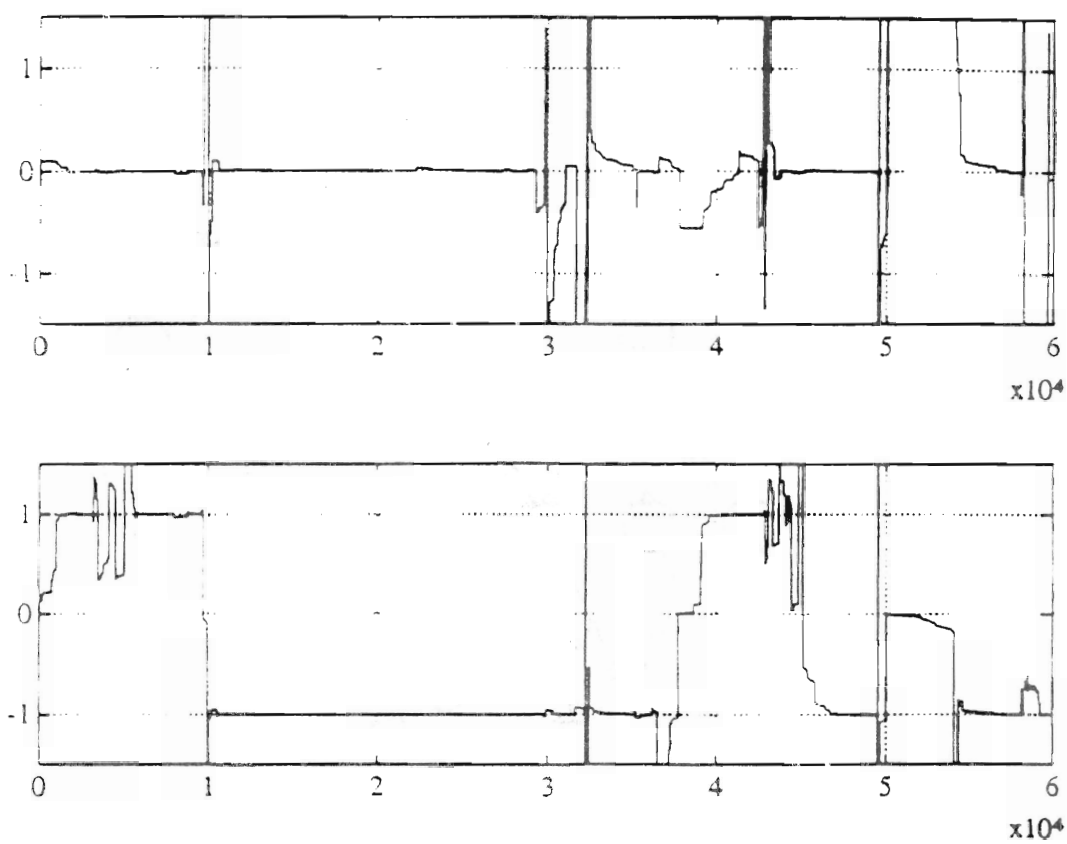


Fig.9 Evolution of the connection strengths W_{11} and W_{12} in a PCA network for the data in Fig.8

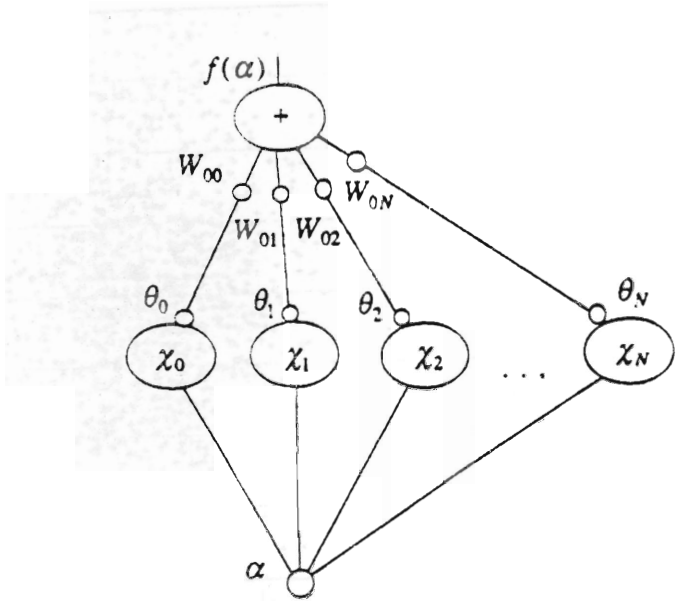


Fig.10 Network to learn the characteristic function of a scalar process

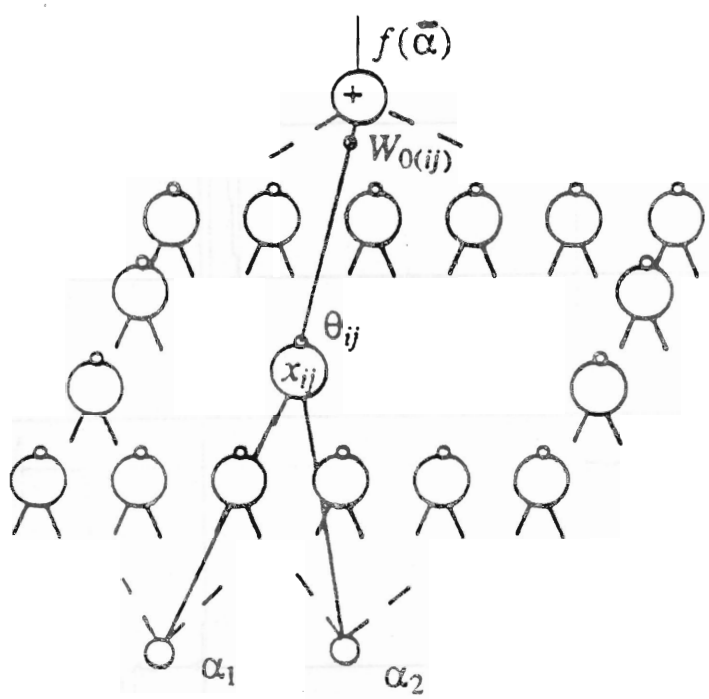


Fig.11 Network to learn the characteristic function of a 2-dimensional signal

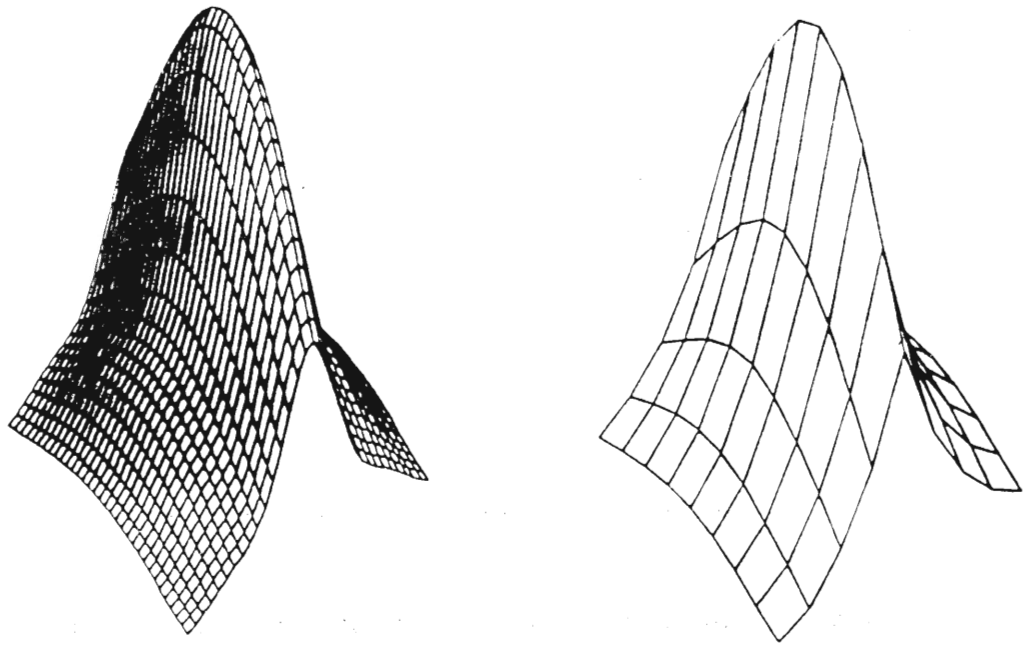


Fig.12a - Real part of the characteristic function for the data in Fig.8 (left) and the mesh of θ_i values (right) obtained by the network.

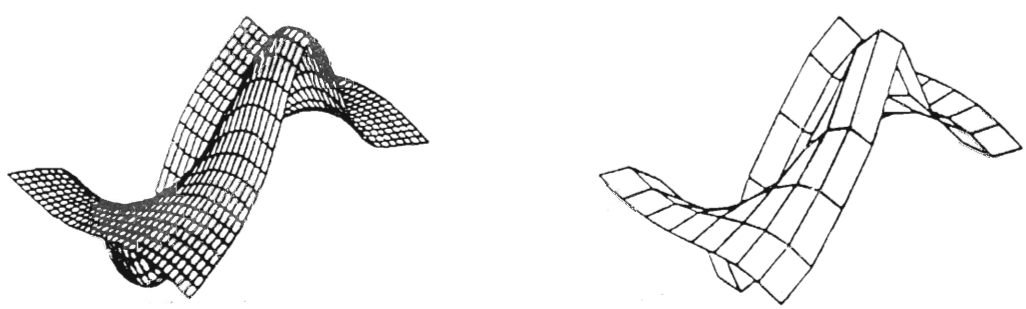


Fig.12b - Imaginary part of the characteristic function for the data in Fig.8 (left) and the mesh of θ_i values (right) obtained by the network.

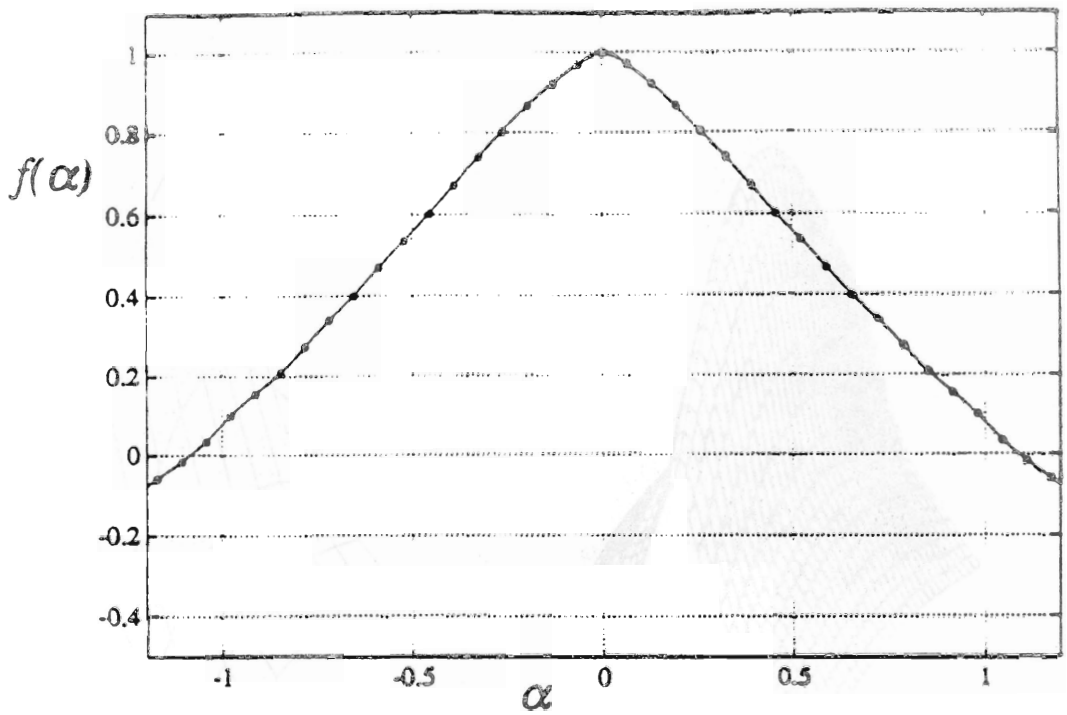


Fig.13a Characteristic function for the Weierstrass random walk (b=1.31)

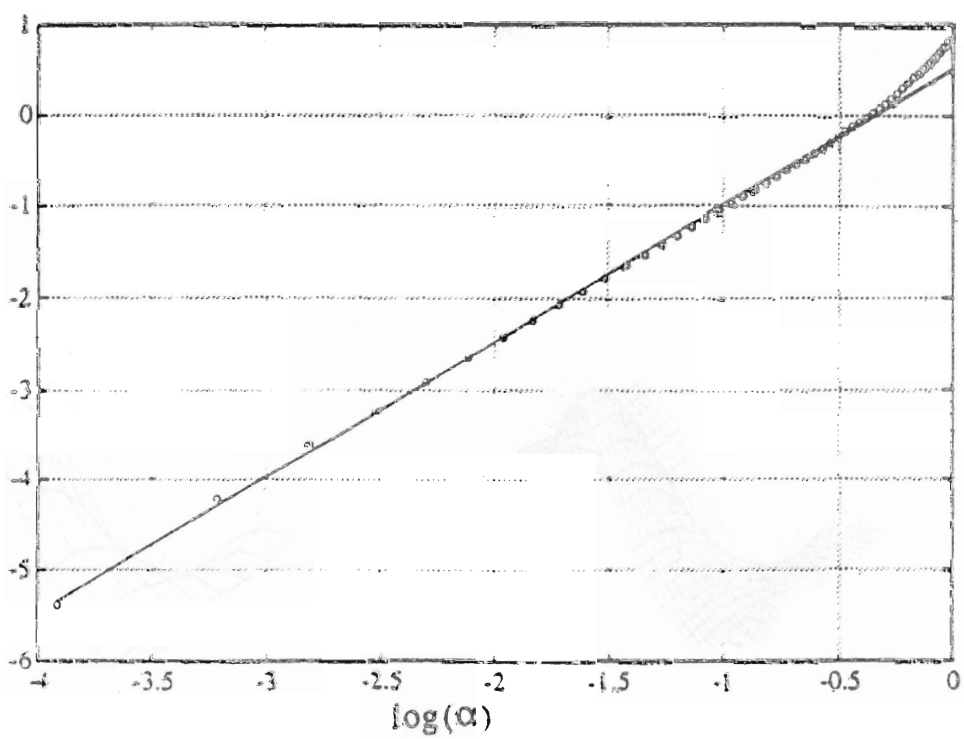


Fig.13b log(-log) of the characteristic function $f(\alpha)$ for the Weierstrass random walk (b=1.31)

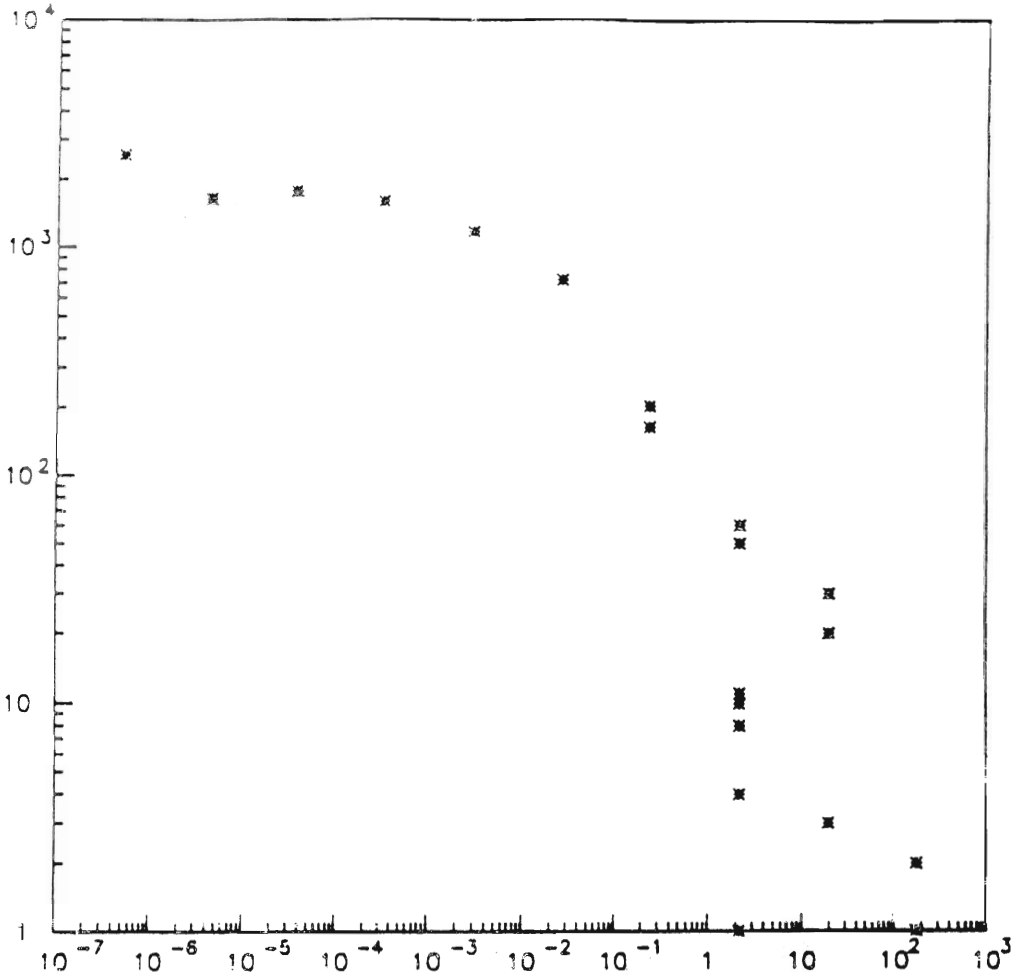
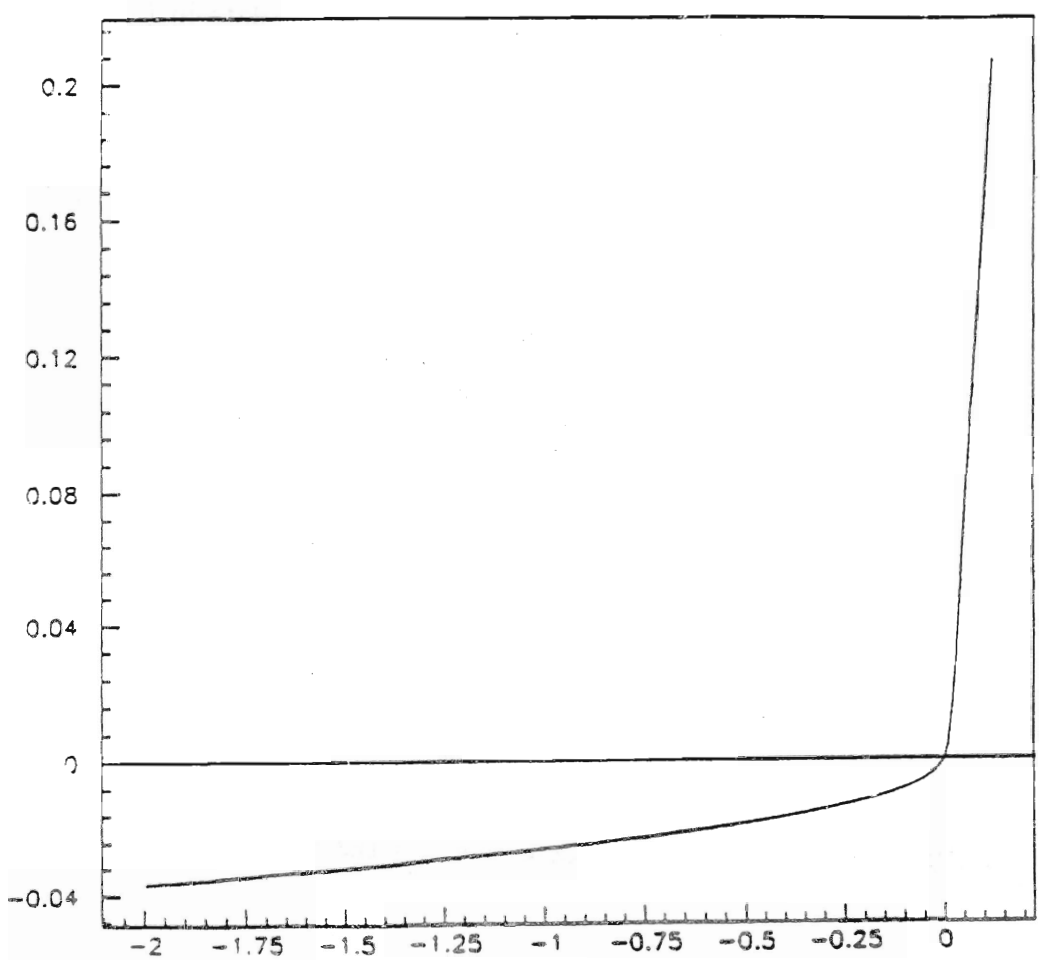


Fig.14 Set of 10^4 events of a multiplicative process with $< X >= 1$. Sample average=0.285. Bin width10^{-6}.



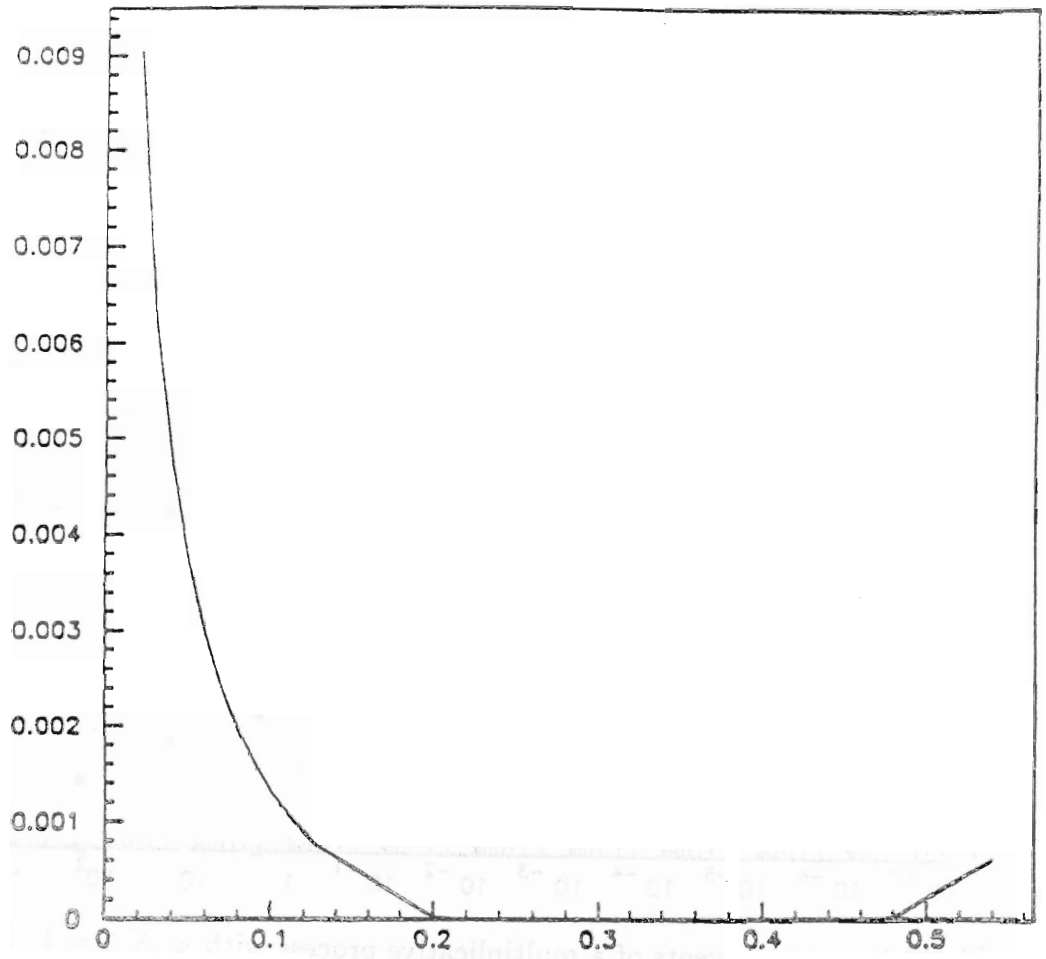


Fig.16 Deviation function estimated from the data in Fig.14

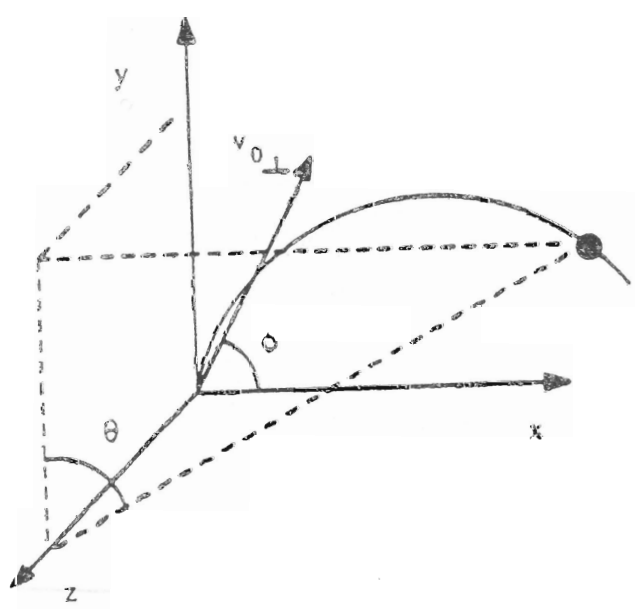


Fig.17 Definition of the variables used for the track reconstruction algorithm

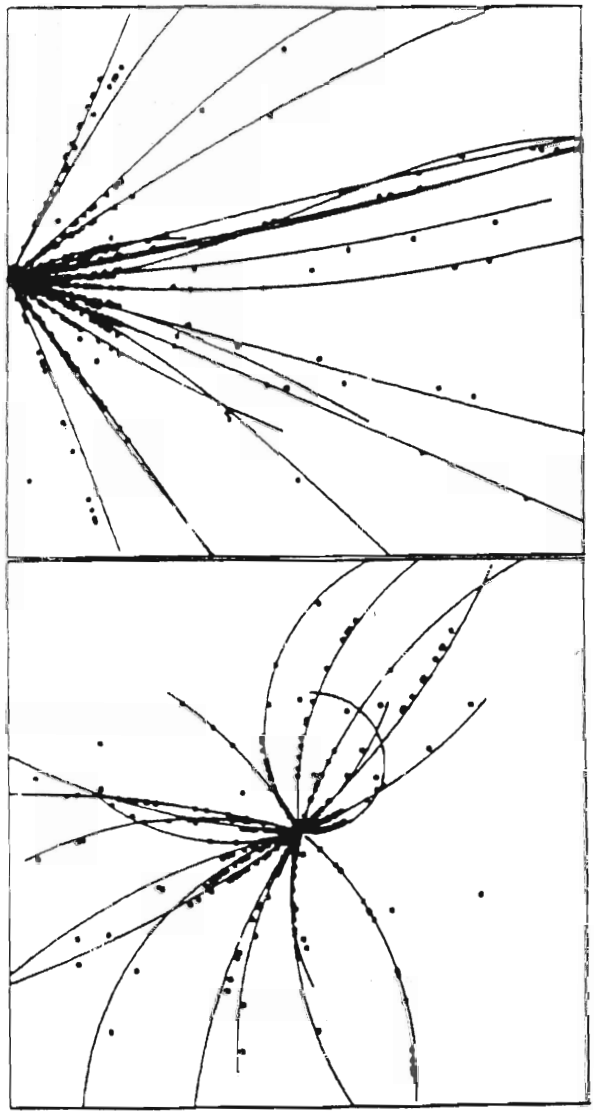


Fig.18 A typical reconstruction example using the track reconstruction algorithm. The upper plot represents the yz projection and the lower plot the xy projection. The dots correspond to the points used for the reconstruction and the full lines to the reconstructed spirals

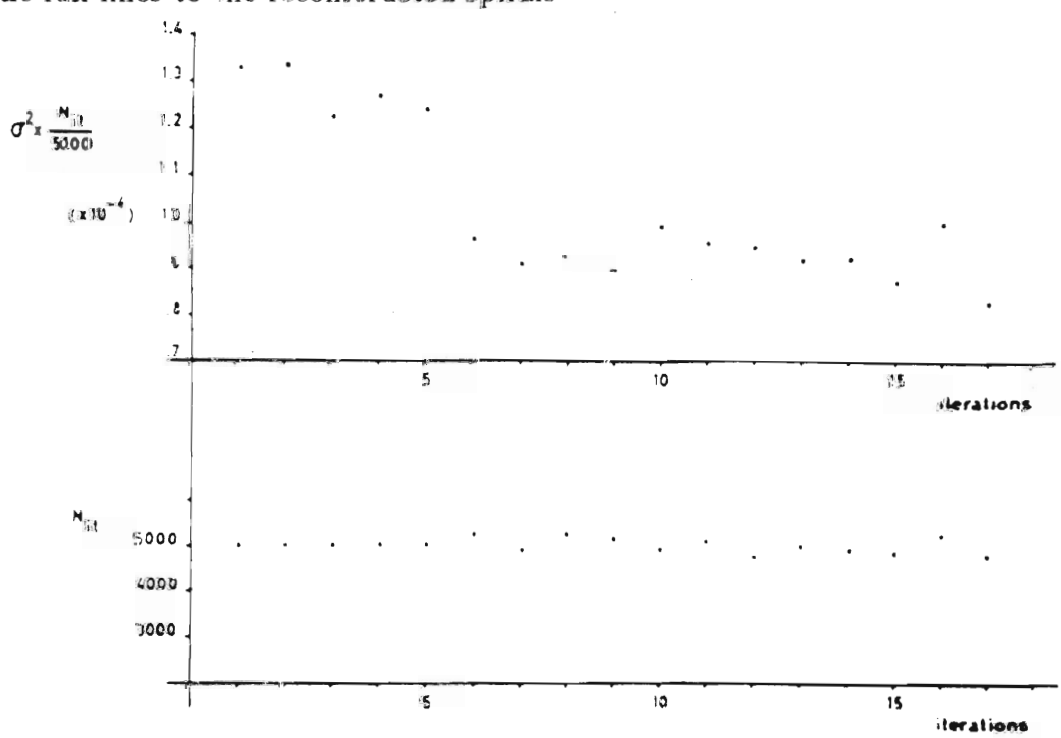


Fig.19 Variance and number of sampled points per iteration for a typical test