

Fermi National Accelerator Laboratory

FERMILAB-Pub-90/197

Smoothing Algorithm for Histograms of One or More Dimensions *

A. Van Ginneken
Fermi National Accelerator Laboratory
P.O. Box 500
Batavia, Illinois 60510

September 1990

* Submitted to Nucl. Instrum. Methods A.



Smoothing Algorithm for Histograms of One or More Dimensions

A. Van Ginneken
Fermi National Accelerator Laboratory*
P. O. Box 500, Batavia, IL 60510

September 1990

Abstract

A smoothing algorithm based on the Bernstein polynomials and their higher dimensional generalizations is presented. The algorithm allows for easy introduction of boundary conditions and other constraints. Some examples are exhibited including a realistic two dimensional case from the Monte Carlo code CASIM.

1 Introduction

Smoothing of statistical information, whether done by eye or by computer, often meets with considerable skepticism. Obviously, the preferred way around statistical problems is to perform a better experiment or, for practitioners of Monte Carlo simulations, to improve one's calculational power and/or technique. However this is not always expedient, or even possible in many instances, particularly in the more observationally oriented sciences. Specific objections to smoothing include the inherent arbitrariness in the choice of procedure (and within a procedure) along with the apprehension that smoothed results may be regarded with more validity if interpreted independent of the raw data.[1] In some sense, the better a smoothing procedure performs the worse it is with regard to this last point. While fully acknowledging such problems, the need for smoothing over statistical irregularities in large arrays of data, under various circumstances and for various

*Fermi National Accelerator Laboratory is operated by Universities Research Association under contract with the US Department of Energy.

reasons, nonetheless persists and this note presents an algorithm for smoothing out histograms including multi-dimensional ones.

The basic approximants of the algorithm proposed here are the Bernstein polynomials [2, 3] and their multidimensional generalizations. They are well known in approximation theory [4] as well as in the theory of probability [5, 6] and in recent years have received considerable attention in 'computer aided geometric design'. [7, 8, 9] At its most basic level the algorithm simply replaces a histogram with a smoothed histogram. With a little extra effort, the value of the function at one or more *points* is obtained. Boundary conditions and other *a priori* information are easy to impose on the smoothed out version.

The specific motivation for this work arises from problems of interpreting outputs of the Monte Carlo program CASIM [10], although one can easily imagine a number of other applications for the algorithm. For convenience, but also as a useful focus, this type of application is kept closely in mind throughout this note in choice of examples and the like. The reader unfamiliar with any of this need not be concerned about being subjected to a detailed treatment of it here but a few words may help achieve a better perspective. Briefly, CASIM is used to study a variety of radiation problems, around high energy accelerators. A typical usage [11] is to calculate radiation dose [12] as a function of location in the general vicinity of an experimental facility. The result of the Monte Carlo is a large array of dose predictions, each one averaged over the volume of a 'bin' or 'cell' which together completely cover the problem geometry. It is recommended, and usually possible, to idealize the problem to one of cylindrical geometry or to a small set of such problems since the reduction from three to two dimensions results in fewer, larger bins and therefore better statistical accuracy of the Monte Carlo results. CASIM is a weighted (vis-a-vis analog) simulation so that typically a very wide range of predicted dose (twenty or more orders of magnitude) results.

Such calculations, even weighted ones, tend to suffer from statistical difficulties, especially at locations far removed from where the particle beam interacts but where dose estimates are still important. This problem may persist even after spending considerable CPU time and using any or all computational tricks at one's disposal. It is the primary aim of the present algorithm to aid in the interpretation of such statistically imperfect output, i.e., following Monte Carlo, normalization, etc., the smoothing procedure is invoked and its results printed or graphed—preferably along with the unsmoothed ones. Customary ways of dealing with the problem, e.g.,

smoothing a dose contour plot graphically (by eye) over its bumps and depressions, in addition to being less accurate, usually require considerable extra work and may lack consistency from one calculation to the next and from one practitioner to the next. A second, distinct, need for smoothing in the CASIM context arises in calculations of energy deposition made in connection with target heating, quenching of superconducting magnets, etc. Typically, one wishes to estimate the *maximum* energy deposition in the target or superconducting magnet. However the Monte Carlo yields only energy deposition averaged over a set of volume bins and, since intra-bin variations can be large, it may be necessary to sharpen the estimate of the maximum energy deposition and of its location. This means to replace all or part of the histogram with a curve or surface.

To avoid possible confusion a few points are emphasized at the outset: (1) it is *not* required that the result be completely 'wrinkle free' or esthetically pleasing, as in certain computer graphics applications. The goal is accuracy with respect to the *true* distribution underlying the data. [13] (2) It is likewise *not* required that the result be in some simple analytical form to facilitate further manipulation or study. The final answer is actually a product of successively generated factors, each of which is itself a very high order polynomial, and its analytical form is ordinarily not kept track of during computation. (3) In the context of calculating radiation dose one should recognize the distinction between *fitting* the Monte Carlo results to a simple expression [14] for ease of interpolation or extrapolation, from the present smoothing algorithm which is much less presumptive. (4) The histogram or array to be smoothed must represent a continuous function. If, for instance in the CASIM example, nuclear interaction densities are calculated, there will be (physically significant) discontinuities across boundaries between different materials and it makes no sense to smooth over them. By contrast, particle flux and dose [12] are expected to be continuous and smoothing is applicable.

Below, in sec. 2, the basic algorithm for one dimensional histograms is described. Remaining in one dimension, introduction of boundary conditions and other modifications is indicated in sec. 3. Generalization to higher dimensions is discussed in sec. 4. Some examples are included in each section along with a specimen from CASIM in sec. 4. Concluding remarks and caveats are in sec. 5.

2 Basic Algorithm

The minimal form of the algorithm replaces a one dimensional histogram with another, smoother, histogram and is the subject of this section. Prior to presenting the basic algorithm some related properties of the Bernstein polynomials are mentioned. The algorithm's iterative character leads to considerations of a stopping criterion. Predicting values of the function representing the histogram at a set of predetermined *points* is discussed next. The section concludes with an example.

2.1 Bernstein Polynomials

The one dimensional n^{th} order Bernstein polynomial of a function, $f(x)$, defined on the interval $[0, 1]$, in conventional notation, is

$$B_n(f; x) = \sum_{k=0}^n f\left(\frac{k}{n}\right) \binom{n}{k} x^k (1-x)^{n-k}. \quad (1)$$

By definition, strict equality is implied at the end points: $B_n(f; 0) = f(0)$ and $B_n(f; 1) = f(1)$. The interval $[0, 1]$ is easily rescaled to any finite range $[a, b]$. In his proof of Weierstrass' Theorem, Bernstein [2] showed that for $f(x)$ bounded on the interval, $\lim_{n \rightarrow \infty} B_n(f; x) = f(x)$ at any x where $f(x)$ is continuous.

Of great importance to their use as stable and smooth approximants are the *variation diminishing* properties of the Bernstein polynomials.[15] Loosely stated this means that for *any* straight line the number of intersections with the graph of $B_n(f; x)$ does not exceed that with the graph of $f(x)$. It can also be shown that the *total variation* of $B_n(f; x)$ is less than that of $f(x)$, where total variation is defined by $\int_{-\infty}^{+\infty} v(f(x) - \gamma) d\gamma$ where v denotes the variation of its argument and the integration is over all reals. Likewise of relevance is the simultaneous approximation of the function and all of its derivatives, i.e., $\lim_{n \rightarrow \infty} B_n(f; x) = f(x)$, $\lim_{n \rightarrow \infty} B'_n(f; x) = f'(x)$, $\lim_{n \rightarrow \infty} B''_n(f; x) = f''(x)$, etc. However, the convergence may be quite slow.[4] Differentiation and integration of the Bernstein polynomials is easily performed and results in compact expressions.

2.2 Algorithm

The histogram $f(x)$ is assumed to have n bins of equal width (Δx) . [16] It is preferred to work with the cumulative distribution which, unlike the

histogram, represent a *function* known at fixed intervals of x

$$F\left(\frac{k}{n}\right) = \sum_{i=0}^k f\left(\frac{i}{n}\right) \Delta x \quad (2)$$

where $\Delta x = 1/n$ on $[0, 1]$, and with the convention $F(0) = 0$. The m^{th} order Bernstein polynomial of F is then

$$B_m(F; x) = \sum_{k=0}^m F\left(\frac{k}{m}\right) \binom{m}{k} x^k (1-x)^{m-k}. \quad (3)$$

Since $B_m(F; 1) = F(1)$ conservation of the total area covered by the histogram is guaranteed. If $F(k/m)$ is non-decreasing, as occurs when all entries in the histograms are non-negative, then the variation diminishing property guarantees $B_m(F; x)$ to be non-decreasing.

Eq. 3 represents the first step of the basic algorithm. A good choice for m here is the number of bins in the histogram, n . In some sense this makes optimal use of the information since omitting data leads to loss of accuracy and possible bias while choosing $m > n$ does not add any new information and would only tend to interpolate the ‘staircase’ character of the histogram. For $m = n$, eq. 3 is typically a great deal smoother than eq. 2 as well as the underlying true distribution. For the next step, the ratios

$$R_1\left(\frac{k}{n}\right) = \frac{F(k/n)}{B_n(F; k/n)} \quad (4)$$

are taken and the Bernstein polynomial of R_1 is obtained:

$$B_n(R_1; x) = \sum_{k=0}^n R_1\left(\frac{k}{n}\right) \binom{n}{k} x^k (1-x)^{n-k}. \quad (5)$$

A more accurate estimate of $F(x)$ is then

$$E_1(F; x) = B_n(F; x) B_n(R_1; x). \quad (6)$$

Note that $E_1(F; 0) = 0$ and $E_1(F; 1) = F(1)$ still hold by the end point matching properties of $B_n(R_1; x)$. However, $E_1(F; x)$ is no longer guaranteed to be non-decreasing and eq. 6 might predict some smoothed bins to be negative even where this does not make physical sense (e.g., negative dose). This can be remedied (see below) by working directly with $f(x)$ instead of $F(x)$.

Compared with eq. 3, the new estimate, eq. 6, is expected to be closer to $F(\mathbf{x})$, e.g., in the least squares sense, but less smooth as quantified, e.g., by the second derivative. The steps eqs. 4–6 can now be iterated.

$$R_2\left(\frac{k}{n}\right) = \frac{F(k/n)}{E_1(F; k/n)} \quad (7)$$

$$B_n(R_2; \mathbf{x}) = \sum_{k=0}^n R_2\left(\frac{k}{n}\right) \binom{n}{k} x^k (1-x)^{n-k}. \quad (8)$$

$$E_2(F; \mathbf{x}) = E_1(F; \mathbf{x}) B_n(R_2; \mathbf{x}) = B_n(F; \mathbf{x}) B_n(R_1; \mathbf{x}) B_n(R_2; \mathbf{x}) \quad (9)$$

until an acceptable solution is obtained. The (smoothed) content of the k^{th} histogram bin after the i^{th} iteration is then

$$e_{i,k} = \frac{E_i(F; [k+1]/n) - E_i(F; k/n)}{\Delta x}. \quad (10)$$

The $B_n(R_i; k/n)$ have smaller total variation than the R_i which means the R_i approach unity during iteration. Also the $B_n(R_i; k/n)$ cannot be identically unity when the R_i are not and thus stop further progress. Since

$$\sum_{k=0}^n \binom{n}{k} x^k (1-x)^{n-k} = 1 \quad (11)$$

for all \mathbf{x} including $\mathbf{x} = k/n (k = 0, 1, \dots)$, and since the

$$\binom{n}{k} x^k (1-x)^{n-k} \quad (12)$$

form a basis for all polynomials of degree n , any solution other than all $R_i(k/n) = 1$ is excluded. This then is the minimal form of the algorithm where the object is to replace a given histogram with another, smoother, histogram. Computationally it is nothing more than a series of matrix multiplications and since the elements can be precalculated, the procedure is very fast. Storage requirements are modest and do not increase as the iteration progresses.

2.3 Stopping Criteria

Perhaps the most straightforward way to decide when to stop the iteration, eqs. 7–9, is at or near the minimum of some *objective function* or cost function which attempts to balance accuracy, which increases with each step,

with decreasing smoothness. Accuracy and smoothness must be quantified in this context. For simplicity, in the examples shown below the objective function, Ω , is always of the type

$$\begin{aligned} \Omega_i = & \sum_{k=0}^n \left[\frac{E_i(k/n) - F(k/n)}{\sigma_k} \right]^2 \\ & + \alpha \left[\frac{E_i(k/n) - 2E_i([k+1]/n) + E_i([k+2]/n)}{E_i([k+1]/n)} \right]^2 \end{aligned} \quad (13)$$

where i refers to the iteration number, F and E_i are from eqs. 2 and 9, respectively, and σ_k is the standard deviation associated with $F(k/n)$. The first set of terms in eq. 13 is related to accuracy in the usual (weighted) least squares sense. The histogram itself, $f(\mathbf{x})$, and its approximant, $e_i(\mathbf{x})$, could be used in eq. 13 instead of $F(\mathbf{x})$ and $E_i(\mathbf{x})$, but the latter provide for smoother behavior of the objective function over the course of the iterations. This is not surprising since this algorithm is formulated directly in terms of $F(\mathbf{x})$ and $E_i(\mathbf{x})$. Another choice might be

$$\int_{\chi^2}^{\infty} P_{\nu}(\mathbf{x}) d\mathbf{x} \quad (14)$$

where χ^2 corresponds to the first sum of eq. 13 and $P_{\nu}(\mathbf{x})$ is the chi-square distribution with ν degrees of freedom. Eq. 14 is perhaps a more appropriate definition of accuracy in this context but it is not clear how ν should change (diminish) with each iteration. The second set of terms relates to smoothness and approximates $\int_0^1 (d^2 E_i(\mathbf{x})/d\mathbf{x}^2) d\mathbf{x}$. The constant α in eq. 13 is to provide 'right' or 'desired' balance between accuracy and smoothness.

Accuracy and smoothness need not be the only ingredients of the objective function and one can bring other criteria into play, e.g., one may discourage certain features in the solution which may be excluded on *a priori* grounds such as negative bin content in a histogram, extraneous peaks or dips, etc. An intriguing approach is via the concept of maximum entropy.[17] For the CASIM application there has not as yet been any systematic search for a satisfactory objective function. What is finally adopted is likely to be dictated as much by practicality as by physical or mathematical argument: simplicity, speed of convergence, etc., with the final details (e.g., α in eq. 13) to be settled mostly by experience. More on objective functions can be found, e.g., in refs. [9, 18, 19, 20]. A thorough analysis of any of this is beyond the scope of the present study. But the choice of an objective

function, while it has important bearings on the outcome of a given calculation, may be treated independently from the rest of the method and in the present note this is limited to what is stated above plus a few remarks in the various examples.

2.4 Estimates at Points

The basic smoothing algorithm calculates a new histogram, i.e., a set of averages of a function over finite x intervals. If the function varies significantly within an interval it may be necessary to sharpen this to estimating its value at a given point. With the notation introduced in eq. 10, let $e_i(x)$ be the smoothed version of the $f(x)$, which represents the original histogram. Let subscript i indicate the number of corrective iterations already performed, $i = 0$ referring to the value of $e(x)$ derived from the Bernstein polynomial of $F(x)$:

$$e_0(x) = B'_m(F; x) = \sum_{k=0}^{m-1} \left[F\left(\frac{k+1}{m}\right) - F\left(\frac{k}{m}\right) \right] \binom{m-1}{k} x^k (1-x)^{m-1-k}. \quad (15)$$

From eq. 6

$$e_1(x) = E'_1(F; x) = B'(F; x)B(R_1; x) + B(F; x)B'(R_1; x) \quad (16)$$

and in general

$$e_i(x) = E'_i(F; x) = E'_{i-1}(F; x)B(R_i; x) + E_{i-1}(F; x)B'(R_i; x) \quad (17)$$

so that, by calculating $B'(R_i; x)$ and $E'_i(F; x)$ a smoothed estimate at any given x is obtained and, when repeated over a sufficiently dense set of x a histogram is replaced by a smooth curve. Examples of this are shown below. As with the basic algorithm, the computation reduces mostly to matrix multiplications and the matrix elements can be precalculated for computer time economy.

2.5 Example

The algorithm of sec. 2.2 is illustrated by choosing a somewhat arbitrary, but easy-to-integrate function to represent the true distribution:

$$f(x) = \sum_{i=1}^3 \frac{a_i}{\sqrt{2\pi b_i}} e^{-\frac{x^2}{2b_i^2}}, \quad (18)$$

i.e., the sum of three Gaussians. To simulate a data set in histogram form, the interval $([0, 1])$, for convenience) is divided into twenty bins, the function is integrated over each bin, and then randomly perturbed by a factor $[1 + r_N(\sigma)]$, where $r_N(\sigma)$ is a normally distributed random number with zero mean and standard deviation σ . This last statement implies that the error in each bin is proportional to its expectation value and represents an idealization of a condition encountered in CASIM type calculations (actually relative error tends to decrease slowly with increasing bin content). Negative values resulting from the random perturbation are replaced by zero. Finally the total contents of the histogram is normalized to the integral over $[0, 1]$ of eq. 18. This is also an idealization of the Monte Carlo results, where the sum over all bins converges to its asymptotic value much faster than suggested from statistical considerations alone, by virtue of the appropriate conservation laws incorporated in the calculation.

The procedure of sec. 2.2 is now applied to the histogram so derived from eq. 18, with a stopping criterion as in sec 2.3. Comparison of the deviations between algorithm predictions and true bin contents, for various σ and for different sets of random deviations, leads to a reasonable choice of α to balance smoothness with accuracy but without attempting to optimize this parameter. In realistic applications the advantage of checking against the true distribution is lost but a similar strategy may make use of surrogate true distributions, e.g., statistically reliable Monte Carlo results or analytical approximations. While the optimum value of a parameter such as α will depend upon the problem, such a determination should usually suffice for a large class of problems. Also, experience from examining the smoothed data may judge whether they are over- or undersmooth and indicate any needed adjustments.

Fig. 1 shows results of the algorithm acting on four histograms derived in the above fashion from eq. 18, with $a_1 = 1, a_2 = 0.1, a_3 = 0.01$ and $b_1 = 0.1, b_2 = 0.3, b_3 = 0.5$ for all, but with different σ , as indicated in the figure.[21] To make the comparison more meaningful, all are generated with the same set of r_N . In addition to the smoothed histogram predicted by the algorithm, fig. 1 also shows a smooth approximant obtained by estimating eq. 18 at a finely spaced set of points. The same stopping criterion (eq. 13) is used for all σ . It is evident from fig. 1 and entirely expected, that the smoothing increasingly deviates from the true distribution with increasing σ . For $\sigma = 0.1$ this deviation is hardly discernible on the scale presented here, while for the larger σ there is a tendency to reflect random fluctuations present in the data. Some of this may be avoidable by making the stopping

function more strongly dependent on σ .

3 Improvements and Variants

This section describes some of the many possible modifications aiming to improve the basic algorithm by including boundary conditions and other *a priori* information. A related property of the Bernstein polynomials: *end point matching*, is discussed below. Next, an example of a boundary condition is explored. The distribution is similar to one of radial energy deposition density (ρ) in cylindrical geometry: the beam is centered on the cylinder axis and so one knows that at a given depth the maximum of ρ occurs on axis:

$$\frac{d\rho}{dr} = 0 \quad \text{at } r = 0. \quad (19)$$

Below, this condition is imposed on the data in two different ways. Next, two constraints of the inequality type, are examined: (1) monotonically decreasing distributions, and (2) distributions which are positive everywhere. Some remarks on ‘fine tuning’ and on weighting conclude this section.

3.1 End Point Matching

The basic algorithm starts from the cumulative distribution and end point matching at $x = 0$ ensures that the integral of $f(x)$ over all x is conserved

$$B(F; 0) = F(0) = \int_0^1 f(x) dx \quad (20)$$

using the convention [21] $F(X) = \int_X^1 f(x) dx$. In the next step, since $R(0) = F(0)/B(F; 0) = 1$, the end point interpolation yields $B(R; 0) = 1$ and therefore $E_1(F; 0) = F(0)$, etc., so that conservation of the integral is maintained throughout. This is of great virtue in many applications including CASIM, where, as already mentioned, the integral is much better known statistically than would be derived by independently combining its component parts.

For the Bernstein polynomials end point matching extends to the derivatives as well.[4] But the same does not hold for the algorithm. For example, for the first derivative, after one iteration

$$E'_1(F; 0) = B'(F; 0)B(R; 0) + B(F; 0)B'(R; 0) = F'(0) + F(0)R'(0) \quad (21)$$

but $R'(0) \neq 0$, in general. However, at the $x = 1$ end, $B(F; 1) = F(1) = 0$, $B'(F; 1) = F'(1) = f(1)$ and after one iteration:

$$E'_1(F; 1) = B'(F; 1)B(R; 1) + B(F; 1)B'(R; 1) = f(1)R(1). \quad (22)$$

Now, $R(1) = B(F; 1)/F(1)$ and hence is undetermined. This may be exploited by fixing $R(1)$ to suit best the application. Two obvious choices are: (1) $R(1) = 1$, which leads to $B(R; 1) = 1$ and by eq. 22 to end point matching. This is used in the above example, as is evident from fig. 1.[21] This choice seems justified only in cases where one has special confidence in $f(1)$. Otherwise it may be more reasonable to set (2) $R(1) = R(\frac{1}{n})$, i.e., equal to its nearest neighbor which decreases the total variation of $R(x)$ and provides for a smoother approximant. This is done in the remaining examples. Other choices are of course possible and it is not claimed that (2) is always the best.

3.2 Boundary Conditions

As stated above, the introduction of boundary conditions is discussed here only for the rather specialized example of a cylindrically symmetric (radial) distribution. The distribution of sec. 2.5 may serve here as well but with this new interpretation. To emphasize this, the independent variable x is replaced by r and f , corresponding to the ordinate of the histogram, becomes ρ , e.g., the *density* of deposited energy. The total energy deposited in the bin $r_i \leq r < r_{i+1}$ is now

$$t_i = \int_{r_i}^{r_{i+1}} \rho(r) 2\pi r dr \quad (23)$$

and the cumulative distribution is [21]

$$P(r_i) = \int_{r_i}^{\infty} \rho(r) 2\pi r dr = \sum_i^n t_j \quad (24)$$

with $P(0)$ representing the total energy deposition. Two different methods are used to introduce the boundary condition and both start from the cumulative distribution.

The first method is similar to that used with splines in computer graphics, (see, e.g., ref. [9]): extra points are added outside of the given boundary and the value of the function at these points is fixed to match the boundary condition(s). In the present example there are *three* conditions to be met

at $r = 0$: (1) $P(0) = \sum_{j=1}^n t_j$ or conservation of total energy deposited, (2) $P'(0) = 0$ which follows from eq. 24, and (3) $\rho'(0) = 0$ or equivalently $P'''(0) = 0$. No condition on $P''(0) [= 2\pi\rho(0)]$ results. Therefore, three equally spaced points ($r = -1/n, -2/n, -3/n$) will be added and the algorithm of sec. 2.2 is carried out over this extended set. The values of P at these points are obtained by solving the equations

$$\begin{aligned} B_{n+3}(P; 0) &= \sum_{j=1}^n t_j \\ B'_{n+3}(P; 0) &= 0 \\ B'''_{n+3}(P; 0) &= 0 \end{aligned} \quad (25)$$

for $P(-1/n)$, $P(-2/n)$, and $P(-3/n)$. In the next iteration the ratios $R(r_i) = P(r_i)/B(P; r_i)$ are obtained for all $r_i \geq 0$ and $R(-1/n)$, $R(-2/n)$, and $R(-3/n)$ are determined (as is easily derived) from conditions almost identical to eqs. 25:

$$\begin{aligned} B_{n+3}(R; 0) &= 1 \\ B'_{n+3}(R; 0) &= 0 \\ B'''_{n+3}(R; 0) &= 0 \end{aligned} \quad (26)$$

and this prevails during all subsequent iterations. Fig. 2 is the result of applying this method to the same set of four histograms as shown in Fig. 1, but reinterpreted as radial distributions. The same stopping criterion is applied and a smooth approximant is likewise generated here using the extended set of (twenty four) points. The interpretation as a radial distribution also affects estimates at points. Since $dP/dr = 2\pi r\rho$ the smoothed estimate of ρ is $B'_{n+3}(P; r)/2\pi r$, to be replaced at $r = 0$ by $B''_{n+3}(P; 0)/2\pi$.

The second method extends the domain of P to $-1 \leq r \leq 1$ by assuming $P(-r) = P(r)$ or, equivalently, $\rho(-r) = \rho(r)$. The algorithm proceeds as in sec. 2.2 but now using a set of $2n-1$ points.[22] The condition $\rho'(0) = 0$ holds as does $P'(0) = 0$ though conservation is not satisfied since $r = 0$ is no longer an end point. When a minimum is encountered in the objective function, $E_i(P; 0)$ is only approximately equal to $P(0)$. However, at least in this example, the approximation is very good and if need be a final normalization factor can restore exact conservation. Results of this method are shown in fig. 3, again for the same set of histograms and with the same objective function.

The first method appears easier to generalize to other types of conditions, e.g., conservation of some of higher moments of the distribution, whereas embedding the problem in a larger one, as in the second method, relies on some inherent symmetry. Where applicable the latter offers a more natural way to introduce the boundary conditions and, perhaps for this reason, has slightly better results. Imposing such conditions should be done only when there is great interest, e.g., in the value of ρ at $r = 0$ or nearby. In addition, the histogram data must be compatible with this, e.g., when large intra-bin variations in ρ near $r = 0$ hide the presence of a maximum at $r = 0$, imposing this condition—which is in the form of a relation among the $P(k/n)$ —will cause the smoothing to be less accurate away from $r = 0$.

3.3 Constraints

One commonly encountered constraint is usually present in CASIM results: the function underlying the data, e.g., ρ above, is monotonically decreasing with r . The easiest way to ensure the final result will be monotonically decreasing is to combine bins in regions where this constraint is violated (due to statistics) until the histogram itself is monotonic. Care must be taken that no gross bias is thereby introduced. Such averaging should not affect the integral of the distribution, and its higher moments only as little as possible. The following rule, while it can be improved upon, is adopted here for simplicity since it yields a monotonic distribution in a single pass without any gross bias: if a bin (upper limit r_i) is less than or equal in value compared to its neighbor at r_{i-1} no action is taken; if it is larger the *surplus* is spread over an area with a radius of $2r_i$. The smoothing algorithm will retain this monotonicity, with at most minor deviations. Fig. 4 shows the result of enforcing this constraint, with the same treatment of the boundary condition as in fig. 2, with which it should be compared.

A second common constraint, rigorously true in CASIM, is that only *positive* ρ make physical sense. Since the densities predicted by the Monte Carlo are nonnegative and an easy way to keep ρ positive everywhere is to work directly with ρ , instead of the cumulative distribution, since (for $0 \leq r \leq 1$) all coefficients of the ρ 's in the Bernstein polynomials are positive as well. This requires an interpolation rule to convert the histogram to a density at $r = 0, \frac{1}{n}, \frac{2}{n}, \dots, 1$ and thus brings extra assumptions into the smoothing. In the present example ρ values are converted assuming exponential dependence on r within each bin and with a slope derived from its nearest neighbors. Fig. 5 shows results of such a smoothing. In addition

to ensuring $\rho \geq 0$ this method has the advantage that, on general grounds, better convergence is expected for the function to which the procedure is applied as compared to its derivative. The results show actually an improvement over fig. 3 which uses the cumulative distribution and the same treatment of the boundary condition.

3.4 Fine Tuning

Near the minimum of the objective function, Ω_i , the smoothed estimate $E_i(\mathbf{x})$ of the function $P(\mathbf{x})$ typically changes slowly between iterations. In many applications, including the CASIM example, the objective function carries itself enough ambiguity to discourage any fine tuning to pinpoint the precise minimum. Instead the calculation is halted when $\Omega_{i+1} > \Omega_i$ and E_i is adopted as the 'solution'. In cases where one has great confidence in the objective function or where large differences exist between successive E_i near the minimum, some interpolation scheme among the last E_i 's could improve matters. A more systematic approach to this involves taking the *Bernstein polynomial of a Bernstein polynomial*.

After k iterations the algorithm produces

$$E_k(\mathbf{x}) = B_n(F; \mathbf{x})B_n(R_1; \mathbf{x}) \dots B_n(R_{k-1}; \mathbf{x})B_n(R_k; \mathbf{x}) \quad (27)$$

and assume now that $\Omega_{k-1} < \Omega_{k-2}$ and $\Omega_{k-1} < \Omega_k$. Assume also that it has been ascertained somehow that the minimum lies between iterations $k-1$ and k . An intermediate solution between the two is

$$E_k^{(1)}(\mathbf{x}) = B_n(F; \mathbf{x})B_n(R_1; \mathbf{x}) \dots B_n(R_{k-1}; \mathbf{x})B_n^{(1)}(R_k; \mathbf{x}), \quad (28)$$

$B_n^{(1)}(R_k; \mathbf{x}) = B_n[B_n(R_k; \hat{\mathbf{x}}); \mathbf{x}]$ is the Bernstein polynomial of $B_n(R_k; \mathbf{x})$ and $\hat{\mathbf{x}}$ serves to distinguish it from \mathbf{x} . If $\Omega_k^{(1)} < \Omega_{k-1}$ then $E_k^{(1)}(\mathbf{x})$ is preferred to $E_{k-1}(\mathbf{x})$. If not, one can try a solution intermediate between $E_{k-1}(\mathbf{x})$ and $E_k^{(1)}(\mathbf{x})$

$$E_k^{(2)}(\mathbf{x}) = B_n(F; \mathbf{x})B_n(R_1; \mathbf{x}) \dots B_n(R_{k-1}; \mathbf{x})B_n^{(2)}(R_k; \mathbf{x}) \quad (29)$$

where $B_n^{(2)}(R_k; \mathbf{x}) = B_n[B_n^{(1)}(R_k; \hat{\mathbf{x}}); \mathbf{x}]$, $\Omega_k^{(2)}$ is compared with $\Omega_k^{(1)}$, and so on. Details of the search procedure aside, it is clear that a more refined solution can be obtained this way. At present this fine tuning is not implemented in CASIM.

3.5 Weighting

One aspect of weighting the data has already been mentioned: in the objective function, eq. 13, each of the squared residuals is weighted by $1/\sigma_j^2$, where σ_j is the error associated with F_j , though clearly other weightings may be tried or preferred. Weighting can also be introduced into the iteration itself by means of the so called ‘rational’ Bernstein polynomials: [7]

$$B_n^{rat}(F; \mathbf{x}) = \frac{\sum w_j F_j b_j^n(\mathbf{x})}{\sum w_j b_j^n(\mathbf{x})} \quad (30)$$

with $b_j^n = \binom{n}{j} x^j (1-x)^{n-j}$, and w_j the weight of F_j . This will cause the algorithm to favor, i.e., shift the answer closer to, data with large w_j . Note that, by eq. 11, when all $w_j = 1$ eq. 30 reverts back to eq. 1. If all $w_j > 0$ the variation diminishing properties apply to the $B_n^{rat}(F; \mathbf{x})$ as well.

The w_j , in addition to their usual role as a measure of how well F_j is known, may also be useful in steering the convergence of the procedure. For example, increasing the w_j at places where the fit is poor will tend to accelerate the decrease of the χ^2 part of the objective function and help reach its minimum in fewer iterations. However care must be taken in exercising such options to avoid introducing systematic bias into the calculation. Other than their direct use in the objective function, weighting has not been used in CASIM thus far.

4 Two Dimensions

As mentioned in the Introduction smoothing in two dimensions is the prime motivation for this study since most CASIM work is done assuming (or adapting to) cylindrical symmetry. Three dimensional calculations are nonetheless an occasional necessity [23] and since the increased dimension usually spells increased statistical difficulty the need for smoothing may be increased as well. Extension to three (or higher) dimensions is not undertaken here because it is straightforward enough to contemplate while a presentation with examples, etc., might easily become more cumbersome than illuminating.

The usual (‘tensor form’) two dimensional generalization of the Bernstein polynomials is

$$B_{mn}(f; \mathbf{x}, \mathbf{y}) = \sum_{k=0}^m \sum_{l=0}^n f\left(\frac{k}{m}, \frac{l}{n}\right) \binom{m}{k} \binom{n}{l} x^k (1-x)^{m-k} y^l (1-y)^{n-l} \quad (31)$$

with obvious extension to higher dimensions. Different types of multidimensional generalizations exist, see, e.g., ref. [3, 5], but are not considered here. The properties which make the Bernstein polynomials attractive in one dimension (see sec. 2.1) are by and large expected to carry over to the higher dimensions though this should not be taken everywhere for granted.[7] The algorithm works exactly the same as in one dimension. Again one may wish to produce a smoothed histogram or one may calculate values at set of points. Boundary conditions and constraints can be applied here as well and this is illustrated below with some examples, which are essentially extensions of the one dimensional cases above. This is followed by an application to a realistic CASIM dose calculation.

4.1 Examples

Analogous to sec. 3.2 a relatively simple and easy-to-integrate expression is adopted to represent a cylindrically symmetric density:

$$\rho(r, z) = \sum_{i=1}^3 a_i \frac{\gamma + z}{\sqrt{2\pi b_i}} e^{-(r^2 + 2b_i^2)(\gamma + z)/2b_i^2}. \quad (32)$$

This might, e.g., mimic the energy density resulting when a proton beam strikes a target. One is mostly interested in estimating the maximum energy density, *where* it occurs in the target, and how it varies with location in the vicinity of the maximum. Since the maximum must occur somewhere on the (beam) axis, smoothing here means to refine the (bin averaged) ρ of a Monte Carlo to point values along the axis and in its immediate vicinity. The boundary condition of sec. 3.2, i.e., $\partial\rho/\partial r = 0$ at $r = 0$, is considered to hold here also.

Preparation of a test sample in the present example is much the same as in sec. 3.2, and based on the same rationale. Eq. 32 is averaged (analytically) over each bin of a 20×20 array to yield its *true* content which is displayed in fig. 6 as a two dimensional histogram ('lego' plot). To each bin is added a randomly varying normally distributed 'error' proportional to its true content and with negative results set to zero. The total summed over all bins is normalized to that calculated from eq. 32. The smoothing procedure is then tested for various values of the relative error σ . Again one can use the cumulative distribution $P(r, z)$, the integral of ρ over r and z , or work directly with ρ . Boundary conditions are introduced similarly: by including in the array additional points (with $r < 0$) to ensure their satisfaction or by

symmetrizing the density (or distribution) with respect to r . The stopping criterion is similar to the one dimensional one (eq. 13) with the square of the second derivative replaced, somewhat arbitrarily, by

$$(\partial^2 \rho / \partial r^2)^2 + 2(\partial^2 \rho / \partial r \partial z)^2 + (\partial^2 \rho / \partial z^2)^2. \quad (33)$$

Three examples are included: (1) using the *cumulative distribution* with additional points at $r < 0$, (2) ditto but with symmetric boundary condition, and (3) using the *density* with the symmetric condition. In the first example, for simplicity, the extra points are calculated only on a row-by-row basis, i.e., as for a one dimensional problem using only the P values of the row (constant z) in which they are located. Only two extra points (per row) are included which ensure that $B'_{n+2}(P; 0) = 0$ and $B'''_{n+2}(P; 0) = 0$. The third condition (see sec. 3.2), would enforce row-by-row conservation of the integral of the density and thereby tend to undo any smoothing achieved along z . Again one could normalize to match the smoothed total over the entire target to the data, but close agreement makes it unnecessary here. Enforcing the other two conditions row-by-row only also appears to work quite well: in all instances ρ is seen to have a maximum at $r = 0$, at least as calculated here on a mesh with $\Delta r = 0.002$. In examples (2) and (3) above conservation of the sum over all bins is always close and the maximum at $r = 0$ follows from symmetry. In (3) linear interpolation is used to convert the histogram to a function.

Figs. 7-10 show some results of these two dimensional tests of eq. 32 for relative errors of $\sigma = 0.1, 0.25, 0.5$ and 1.0 . For each case a histogram is shown of the random data sample derived from the histogram of fig. 6 along with the smoothed versions obtained using the algorithm as in (1)-(3), above. These figures are only meant to convey a general impression so no scales are included. In figs. 6-10 ρ is plotted on a linear scale which extends from zero to ~ 30 . Because of fluctuations, the data sample plot has occasional excursions beyond this and the ρ scale of figs. 9a and 10a is therefore extended. Plots (a)-(c) of figs. 7-10 are true two dimensional histograms but (d) is a set of ρ values smoothly interpolated, along lines of constant r and of constant z separately, by the graphics program TOPDRAWER.[24]

It is interesting to compare point estimates along the $r = 0$ axis since the maximum of ρ is expected to fall on the axis. These point estimates are the direct result of method (3). When using the cumulative distribution the procedure is much the same as for the one dimensional radial distribution with $e(r, z) = (2\pi r)^{-1} \partial^2 B(P; r, z) / \partial r \partial z$ being the estimate for ρ for $r \neq$

0 and $e(0, z) = (2\pi)^{-1} \partial^3 B(P; 0, z) / \partial^2 r \partial z$. Fig. 11 shows the curves at $r = 0$ for examples (1)-(3) with different σ , along with the actual values from eq. 32. Also shown is a histogram of the string of bins closest to $r = 0$ ($r \leq 0.05$) although this represents only a small part of the data used in the smoothing at $r = 0$ and averaging over $r \leq 0.05$ introduces a slight downward bias. Observations made for the one dimensional examples seem to remain valid: using the density works best and, when using the cumulative distribution, the symmetric boundary condition performs better. This makes sense for reasons referred to in the one dimensional case though one cannot draw any firm conclusions based on this limited sampling.

4.2 CASIM Application

The typical CASIM calculation results in a 50×50 two dimensional histogram of dose predictions as a function of location. To implement what can be concluded from the above examples into a procedure, a good choice appears to be: (1) convert the histogram to a function on an equispaced grid prior to smoothing, (2) include the constraint of a monotonical decrease with radius at constant z , (3) ignore the boundary conditions at $r = 0$, since one is mostly interested in the dose at large radii. It is often possible, and helps the smoothing, to add another constraint: (4) at constant r , the z distribution is restricted to have (at most) a single maximum.[25] In realistic applications this constraint may have more exceptions to it than the radial one.

Typically, the dose predicted by such a calculation varies enormously with location (twenty orders of magnitude in the example below), which implies large intra-bin variations. Therefore, care must be taken in the point interpolation from the histograms.[26] Also because of this enormous variation, the first step of the algorithm is performed on $\log \rho$, but continues with linear ratios thereafter. To accomodate taking logarithms, bins with zero content are augmented by a small dose (ϵ_D) equal to one tenth of the smallest of non zero bin. The objective function is as for the examples of sec. 4.1. The actual errors estimated from the Monte Carlo are used in the 'least squares' term [27] but, since the *relative* error varies only slowly with location, Ω remains sensitive even to very low dose predictions at large r . The curvature term varies more smoothly from one iteration to the next if one uses the sum of the squares of the principal curvatures of the surface in lieu of eq. 33. Fig. 12 shows the raw histogram of a realistic CASIM calculation along with the smoothed version, obtained after three iterations.

In fig. 12 the ρ scale is logarithmic and spans some twenty orders of magnitude. The ‘floor’ consists of all bins or points below ϵ_D . As can be seen a reasonably smooth solution is obtained. Some wrinkles, e.g., the ones at large z and small r are probably removable by weighting since statistics is very poor in that region. Fine tuning (sec. 3.4) is possibly of significant benefit here.

Estimates of dose in the region upstream of the target, i.e., due to particles traveling opposite to the beam direction, are obtained in CASIM by the artifice of allowing the beam to penetrate the first five bins along z without interaction. Especially at high energy this results in a large difference between the fifth and sixth bin along z , near $r = 0$. Fig. 13 shows an enlargement of a region of 10×10 bins at the beginning and center of the target. The ρ scale is again logarithmic, and along the $r = 0$ axis, the sixth z -bin is larger by a factor of ~ 150 than the fifth. The agreement which is typically within the Monte Carlo error in this general vicinity, is no better than a factor of $2 - 3$ at the jump itself, but one should note that (a) the interpolation procedure from histogram to function plays a role in this, and (b) with more iterations a better approximation would result in this region, at the expense of introducing more structure (due to fluctuations) in the solution elsewhere. The lesson is probably to ignore the smoothing in this region if one wishes to concentrate on dose at large r . If, instead, one seeks a smooth approximant in the region of fig. 13 one might succeed better repeating the calculation over a smaller patch of z and r .

5 Concluding Remarks

Much work remains to be done if the algorithm is to be placed on firm footing. The few examples above show some promise but offer no assurance for more general applicability and, emphatically, no such claims are made here. A larger variety of such examples has to be examined, including cases with more inherent structure such as single or multiple peaks of various shapes, large single bin fluctuations, etc. As part of such studies there should be a comparison with existing smoothing algorithms. There is a need to be more quantitative in how well the smoothing performs: a useful addition would be the inclusion of an error analysis to help judge the merits of different procedures applied to the same data. This could also be useful in studying stopping criteria, which as already amply referred to above, must be sharpened.

Bearing in mind these caveats and those connected with smoothing in general, the algorithm described above is easy to implement and appears capable of yielding a smooth approximant to an equispaced histogram of one or more dimensions. Boundary conditions and constraints of the inequality type are easy to include. Derivatives and moments are asymptotically reproduced. Following some preparation the operations become mostly repeated matrix multiplications, using the same matrix throughout. The matrix elements are themselves easy to calculate and can, if desired, be precalculated for repeated later usage. The algorithm is reasonably fast. The CASIM dose application is by far the longest in duration of the above examples. Without any effort to economize it runs in about 80 seconds of CPU time on a CYBER 875 including precalculation of all matrix elements and actually most of this time is spent on calculating logarithms and exponentials, which is incidental to this particular application and not inherent in the algorithm. The 80 seconds is to be compared with about one hour of CPU time to run the Monte Carlo and so it seems well worthwhile to include the smoothing. Possible further improvements have been mentioned in the sections on fine tuning and on weighting. It would also be of interest to examine the possibilities of adapting the algorithm to *extrapolation*. At some higher level of sophistication one might envision some true hybrid which combines smoothing and fitting through adroit use of boundary conditions, etc.

My thanks to D. Finley and L. Michelotti for their comments.

References

- [1] Throughout, 'data' refers to both experimental and Monte Carlo results, although the latter usage is somewhat dubious.
- [2] S. Bernstein, *Commun. Soc. Math. Kharkow* (2), 13 (1912-13) 1.
- [3] G. Lorentz, *Bernstein Polynomials* (Univ. Toronto Press, Toronto, 1953; 2nd pr., Chelsea, New York, 1987).
- [4] P. J. Davis, *Interpolation and Approximation*, (Blaisdell, Waltham, MA 1963; 2nd pr., Dover, New York, 1975).
- [5] W. Feller, *An Introduction to Probability Theory and its Applications*, Vol. II, (J. Wiley & Sons, New York, 1966).

- [6] M. Kac, *Statistical Independence in Probability, Analysis, and Number Theory*, (Mathematical Association of America, Washington, D.C., 1959).
- [7] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design. A Practical Guide*, (Academic Press, San Diego, 1988) provides an introduction to 'computer aided geometric design' including its use of Bernstein polynomials. This subject generally deals with *parametric* curves and surfaces, not *functions* as is the case here and in most of the mathematics literature. For this reason the latter terminology and notation adopted here. Some differences exist in nomenclature between the two, e.g., eq. 1, with x being a parameter, is called the Bézier curve of f by Farin et al., whereas the factors multiplying the $f(\frac{k}{n})$ are referred to instead as Bernstein polynomials.
- [8] T. Lyche and L. L. Schumaker, Eds., *Mathematical Methods in Computer Aided Geometric Design*, (Academic Press, Boston, 1988).
- [9] Su Bu-Qing and Liu Ding-Yuan, *Computational Geometry*, (Academic Press, San Diego, 1989).
- [10] A. Van Ginneken, CASIM. Program to Simulate Hadronic Cascades in Bulk Matter, Fermilab FN-272 (1975).
- [11] J. D. Cossairt et al., *Nucl. Inst. Meth.*, 197 (1982) 465; J. D. Cossairt et al, *ibid.*, A238 (1985) 504; N. V. Mokhov and J. D. Cossairt, *ibid.*, A244 (1986) 349.
- [12] 'Dose' refers here to dose delivered by the radiation present at a given location and as measured, e.g., by its biological impact (not to energy absorbed by the material which actually occupies that location).
- [13] This goal represents an idealization unreachable in practice due to systematic errors, lack of resolution, etc. For practical purposes it should be possible in a given individual case to sort out what can and cannot be achieved by smoothing.
- [14] B. J. Moyer, *Evaluation of Shielding Required for the Improved Bevatron*, Lawrence Radiation Lab. Report UCRL-9769 (1961).
- [15] I.J. Schoenberg, in: *Numerical Approximation*, ed. R. E. Langer (University of Wisconsin Press, Madison, 1959).

- [16] As presently formulated, equal bin widths are essential in the algorithm and, where necessary, data must be re-binned to conform.
- [17] see, e.g., C. Fuglesang, Nucl. Inst. Meth. A278 (1989) 765.
- [18] P. Lancaster and K. Salkauskas, Curve and Surface Fitting, (Academic Press, San Diego, 1986).
- [19] G. Wahba, J. Roy. Stat. Soc., Ser. B, 40 (1978) 364; P. Craven and G. Wahba, Numerische Mathematik, 31 (1979) 377.
- [20] R. Kohn and C. F. Ansley, SIAM J. Sci. Stat. Comput., 8 (1987) 33.
- [21] Because eq. 18 strongly peaks at $z = 0$, in calculating the results of fig. 1, and in the examples below as well, the cumulative distribution is calculated from the outside in, i.e., starting at $z = 1$, to avoid errors associated with adding small contributions to a large cumulative total.
- [22] By exploiting the symmetry of the problem, there is no significant increase in CPU time or memory requirements compared with sec. 2.2. Such considerations become more meaningful when working in two or more dimensions.
- [23] see, e.g., A. Van Ginneken, D. Edwards, and M. Harrison, in Physics of Particle Accelerators, Vol II, p.2033, M. Month and M. Dienes, Eds., (AIP, New York, 1989).
- [24] J. Clement, TOPDRAWER. Bonner Lab Version, Rice University preprint, (May 1990). The program uses a smoothing routine from J. W. Tukey, Exploratory Data Analysis, (Addison-Wesley, Reading, MA, 1977). All graphs in the present note are prepared with TOPDRAWER.
- [25] Actual implementation is somewhat looser: at each r the centroid, z_0 , is determined. Outside of ± 10 binwidths of z_0 the distribution is made monotonically increasing with z for $z < z_0$, decreasing for $z > z_0$, by averaging out the surplus of errant bins. Since the maximum could lie outside ± 10 bins of z_0 , the process works inward—towards the peak. If, outside 10 bins from z_0 , a bin has contents larger than the average of all 50 bins, at that r , then the constraint is not enforced beyond that point, on that side of the maximum.

- [26] Within a bin, dose is assumed to vary exponentially with r and with z and slopes as determined from averaging rates of change over nearby bins. The dose at a point of intersection of four bins is taken as the average of the (four) doses needed to reproduce the CASIM prediction in each bin, given the averaged local slopes. (At the edges this reduces to two, at the corners to a single bin.) The overall total is recalculated from the doses obtained by point interpolation, again assuming exponential dependence on r and z , and an overall normalization is applied to reproduce the total CASIM prediction. This last step is repeated in subsequent iterations.
- [27] Use of the least squares term as in eq. 13 is somewhat simplistic since it ignores large correlations known to exist among neighboring bins in all such calculations.

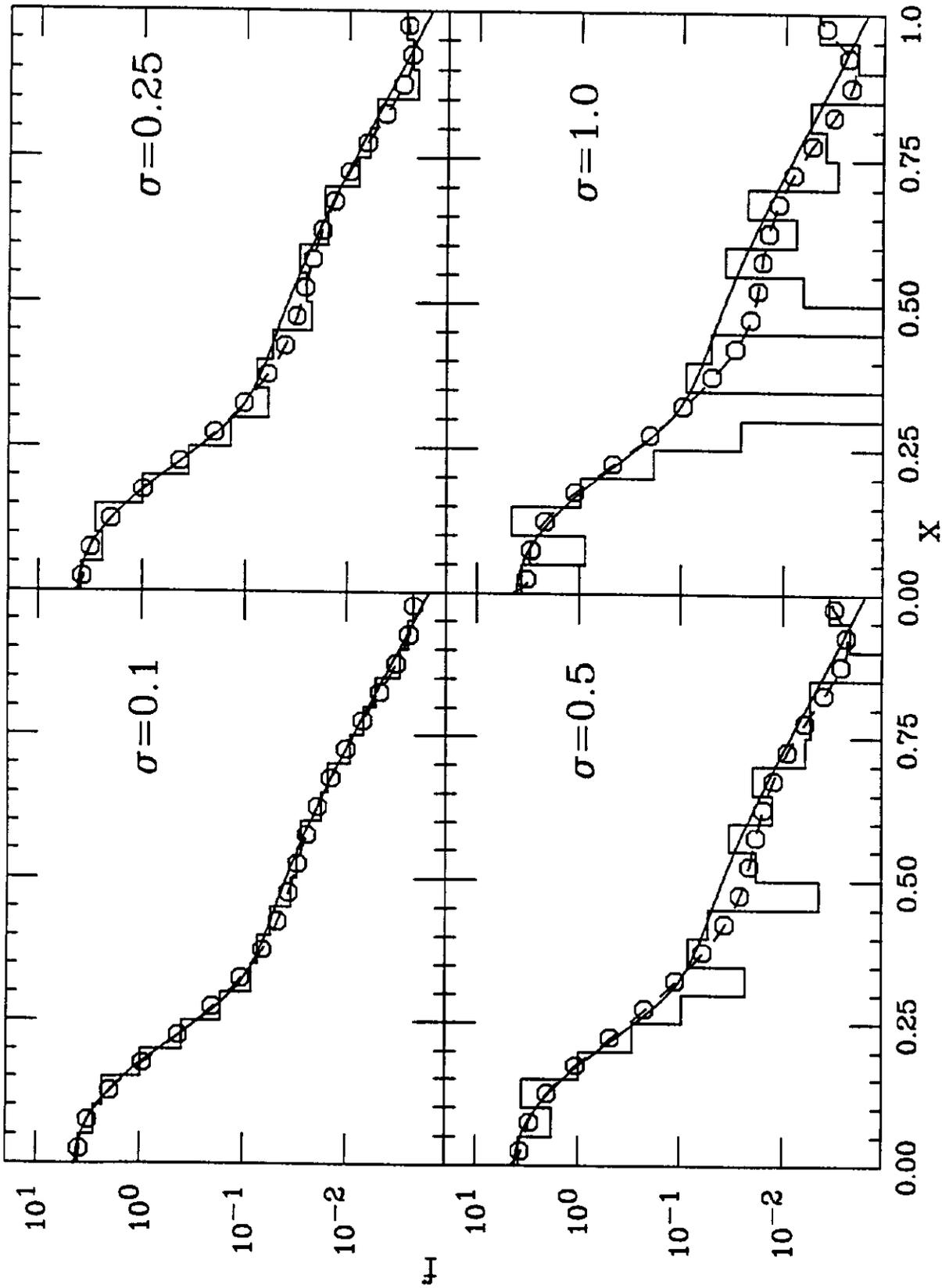


Fig. 1 Test of plain algorithm on histogram based on eq. 18 (solid curve) for selected relative errors (σ); \square are averages over bin, dashed curve is continuous approximation.

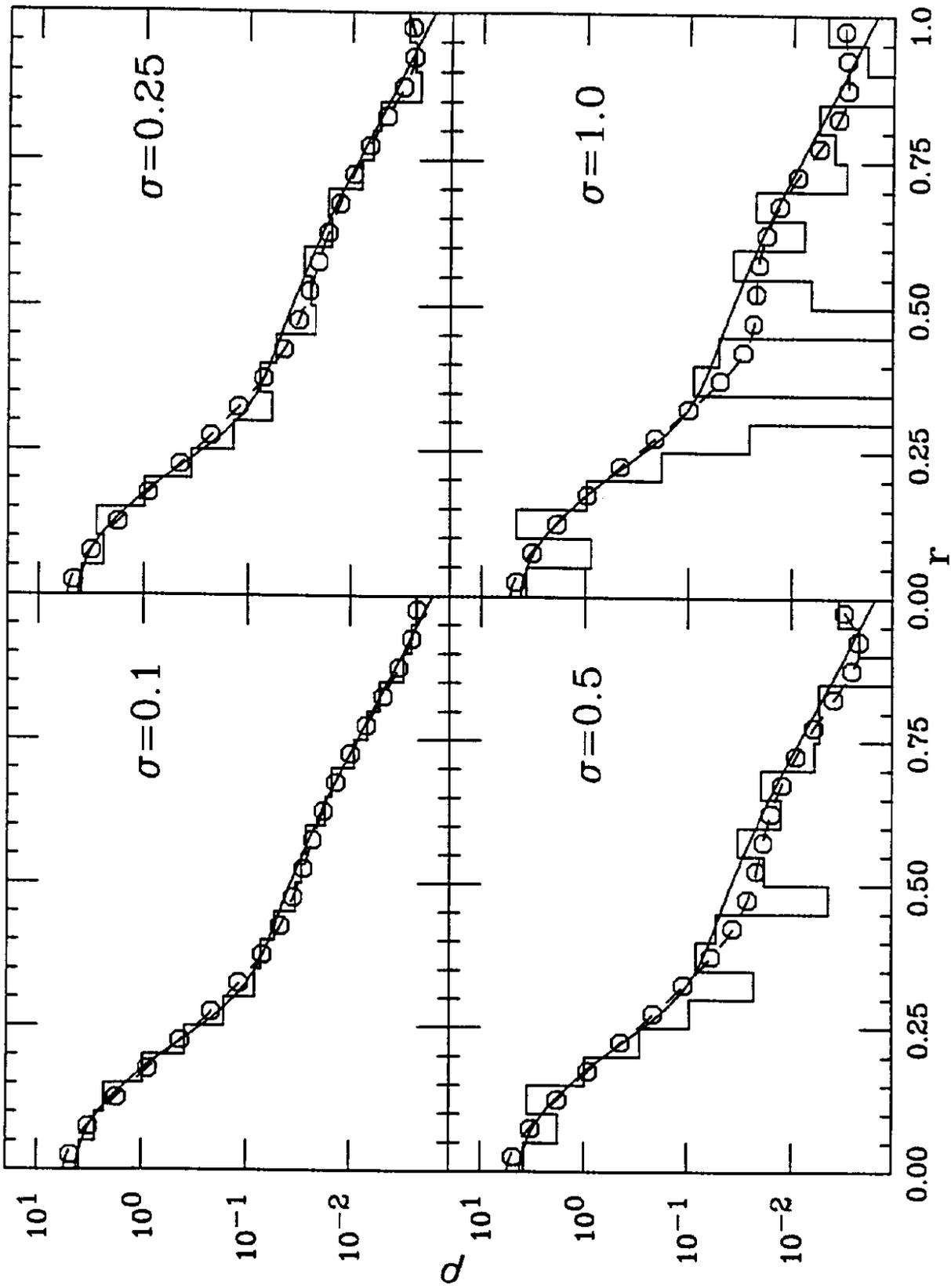


Fig. 2 Test of algorithm on histogram based on eq. 18 (solid curve) interpreted as a radial distribution, for selected relative errors (σ); boundary condition, eq. 19, is applied by adding extra points at $r < 0$; \circ are averages over bin, dashed curve is continuous approximation.

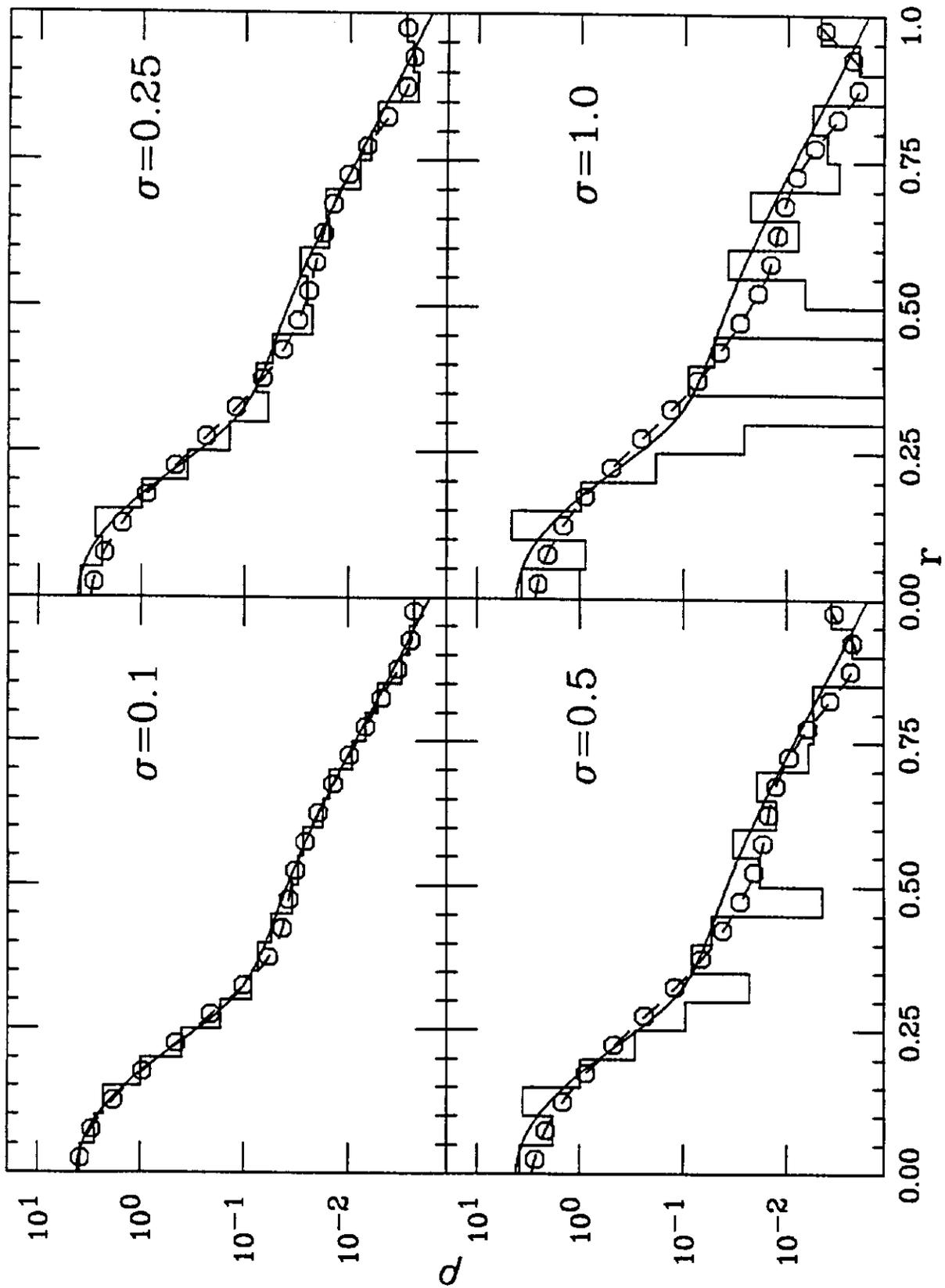


Fig. 3 Test of algorithm on histogram based on eq. 18 (solid curve) interpreted as a radial distribution, for selected relative errors (σ); boundary condition, eq. 19, is applied by symmetry with respect to $r = 0$; \circ are averages over bin, dashed curve is continuous approximation.

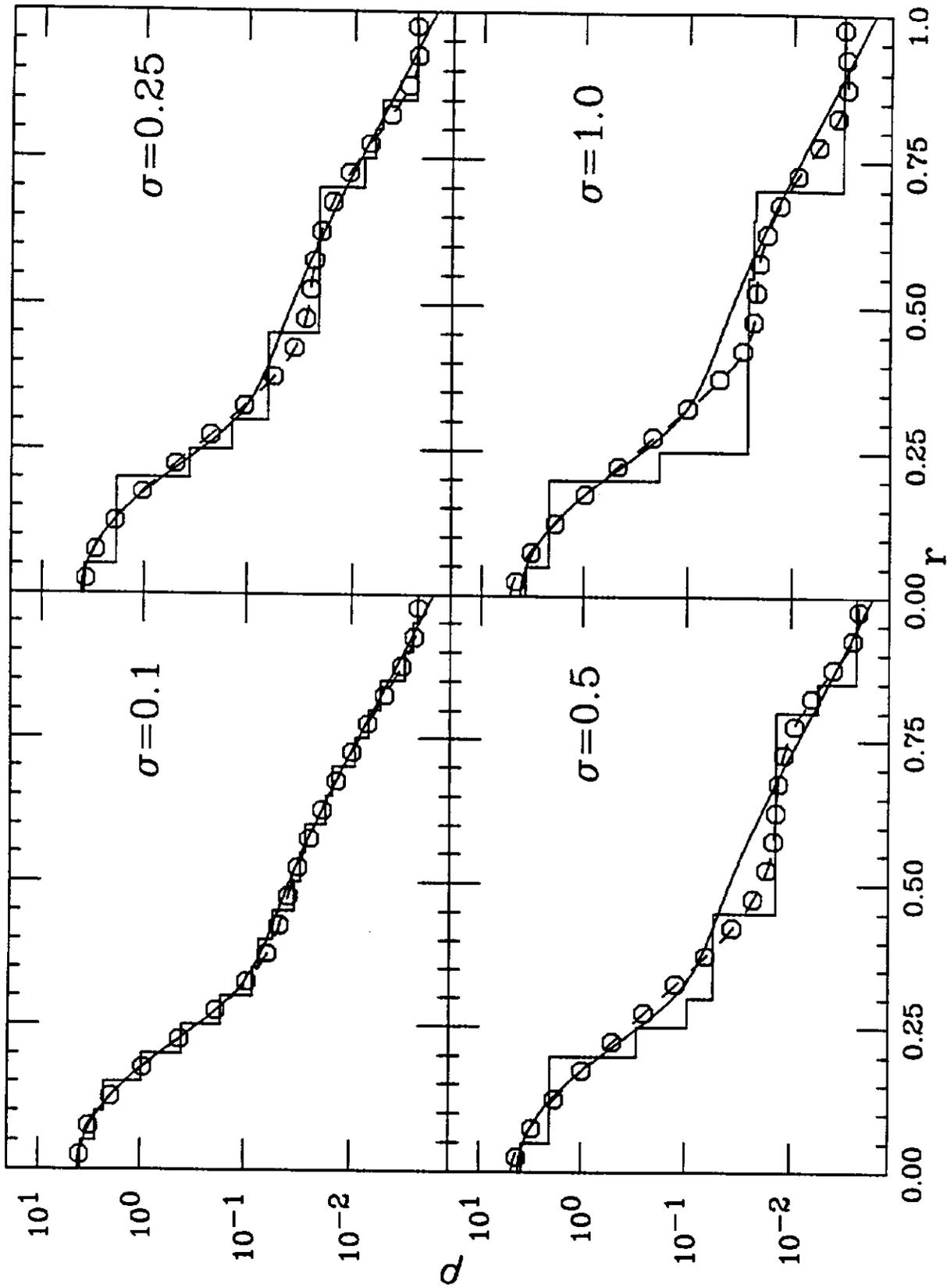


Fig. 4 Test of algorithm on histogram based on eq. 18 (solid curve) interpreted as a radial distribution, for selected relative errors (σ); boundary condition, eq. 19, is applied by adding extra points at $r < 0$ and constraint of monotonic decrease with r is imposed; \circ are averages over bin, dashed curve is continuous approximation.

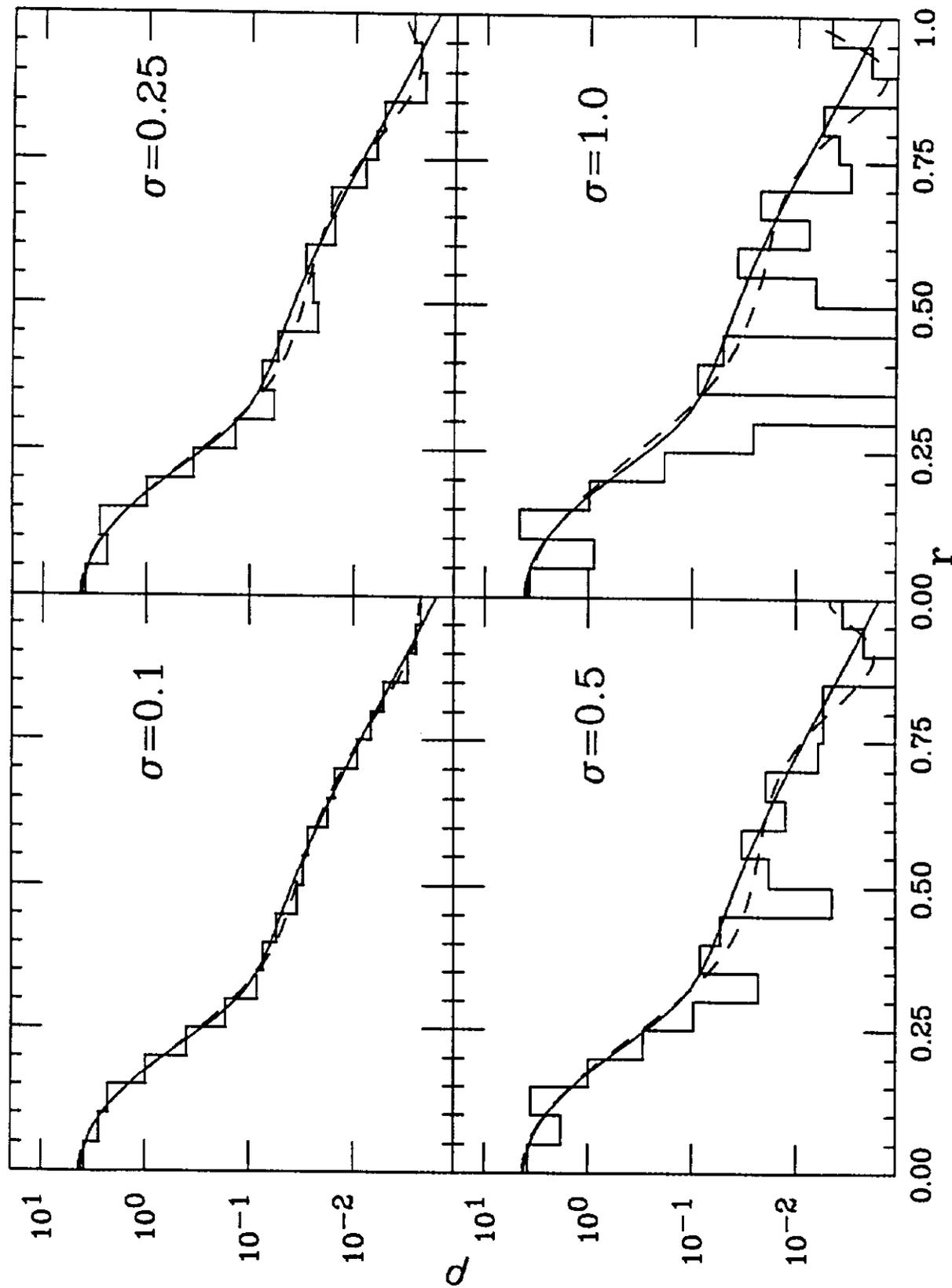


Fig. 5 Test of algorithm on histogram based on eq. 18 (solid curve) interpreted as a radial distribution, for selected relative errors (σ); histogram is converted to function using logarithmic interpolation; boundary condition, eq. 19, is applied by symmetry with respect to $r = 0$; dashed curve is continuous approximation.

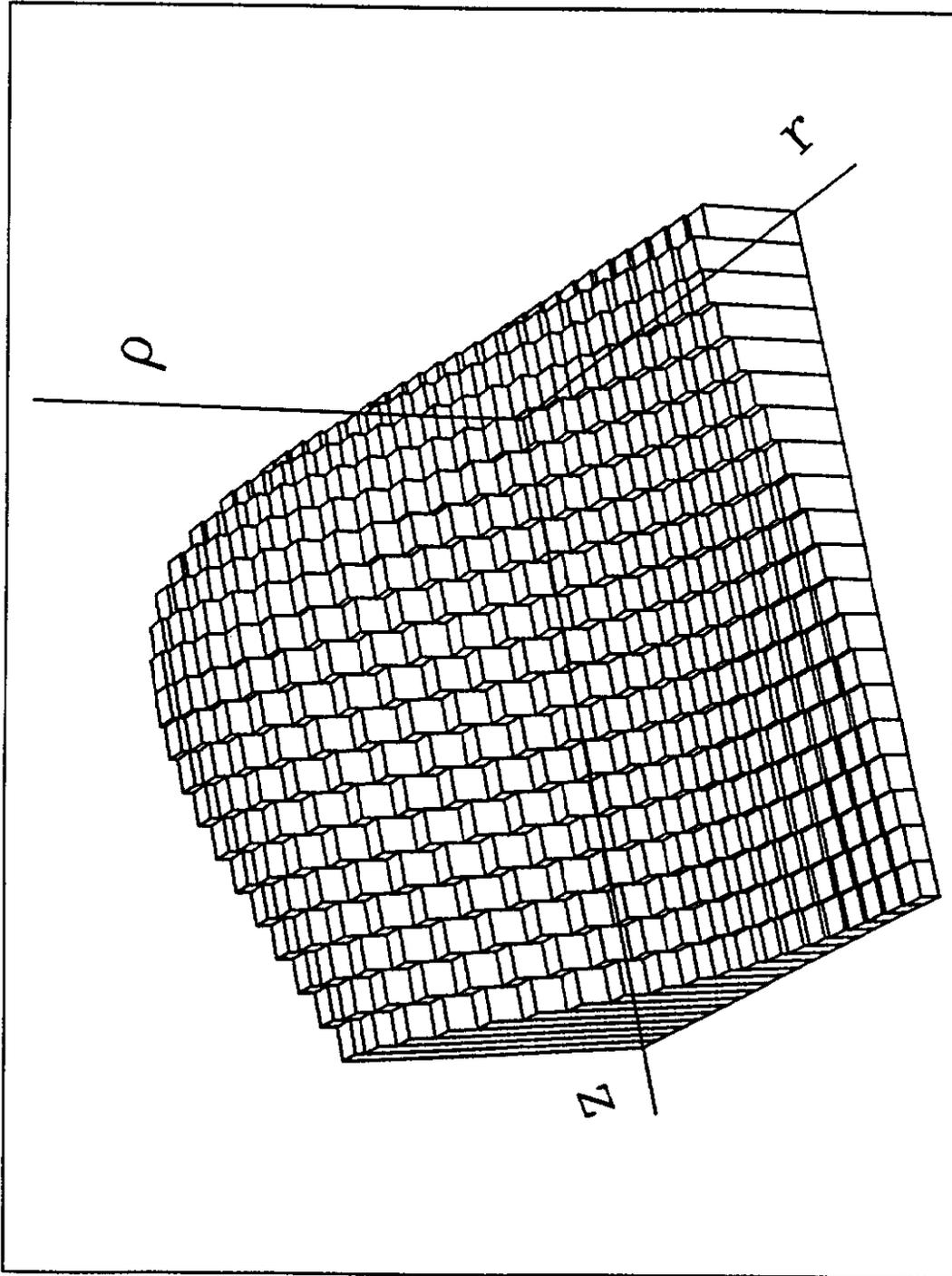


Fig. 6 Two dimensional histogram derived from eq. 32.

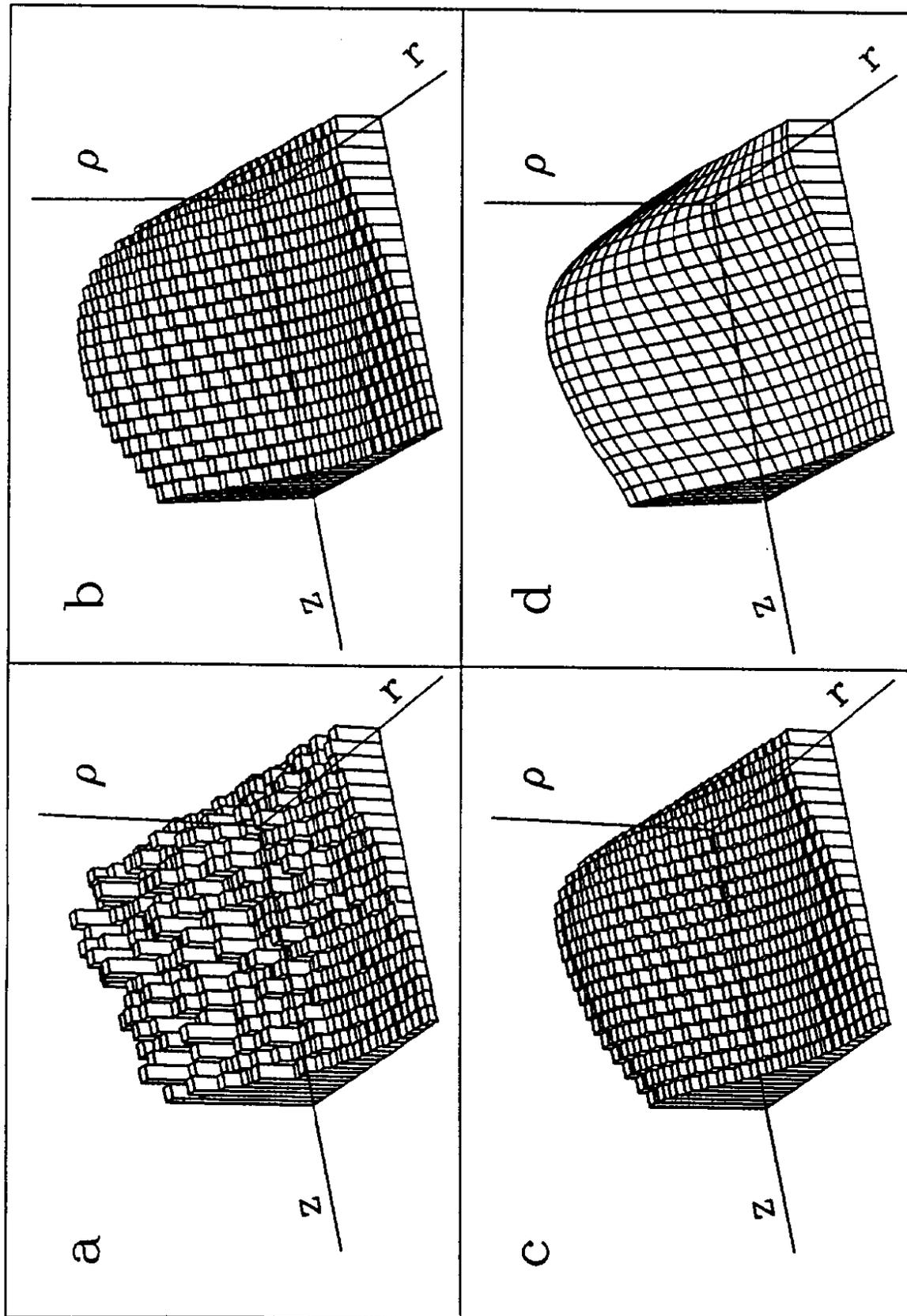


Fig. 7 Test of algorithm on (a) histogram based on eq. 32 with relative error $\sigma = 0.10$. Predictions of algorithm with boundary condition eq. 19 applied using (b) extra points at $r < 0$, (c) symmetry with respect to $r = 0$, (d) linear interpolation on equispaced grid and symmetry with respect to $r = 0$.

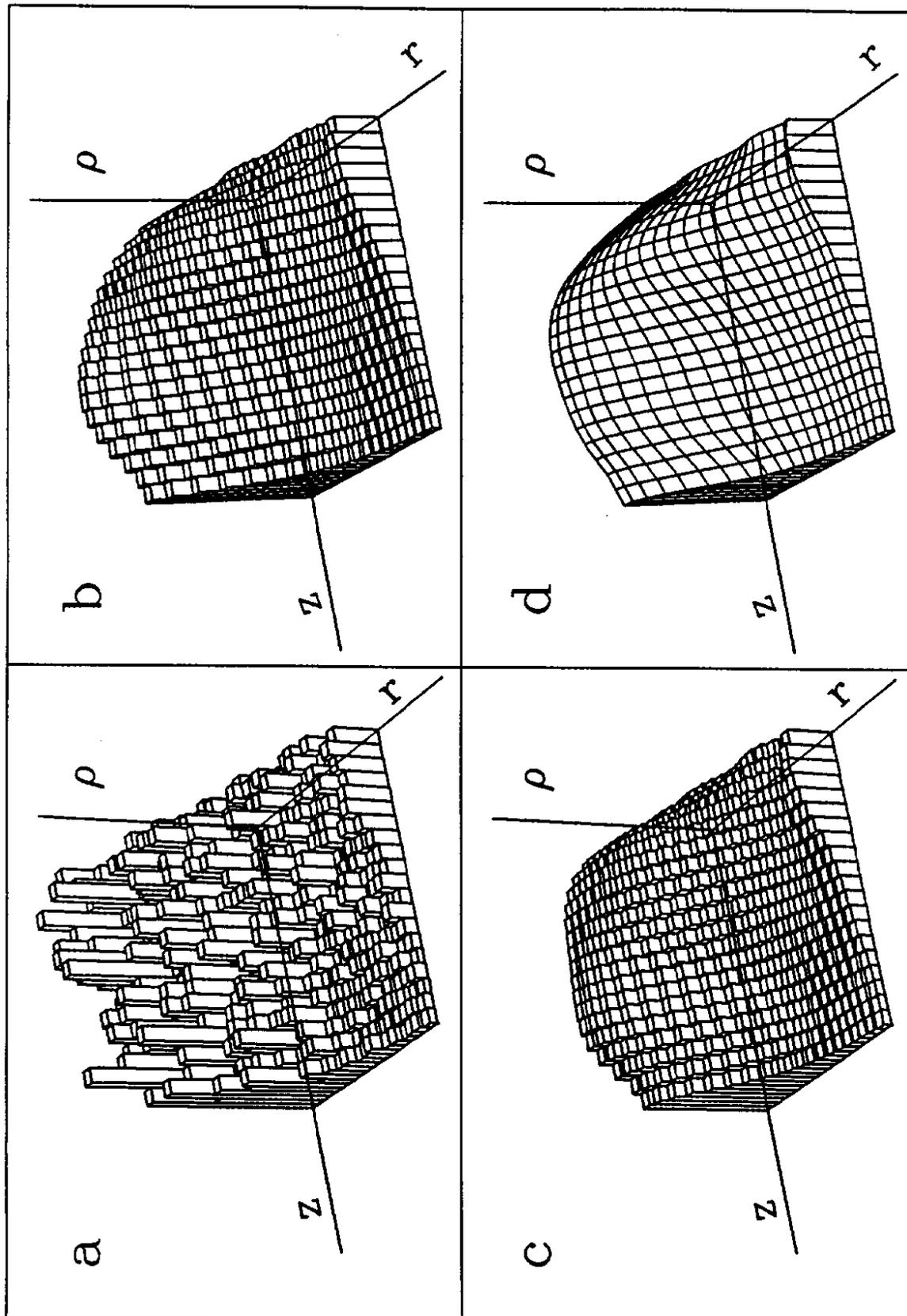


Fig. 8 Test of algorithm on (a) histogram based on eq. 32 with relative error $\sigma = 0.25$. Predictions of algorithm with boundary condition eq. 19 applied using (b) extra points at $r < 0$, (c) symmetry with respect to $r = 0$, (d) linear interpolation on equispaced grid and symmetry with respect to $r = 0$.

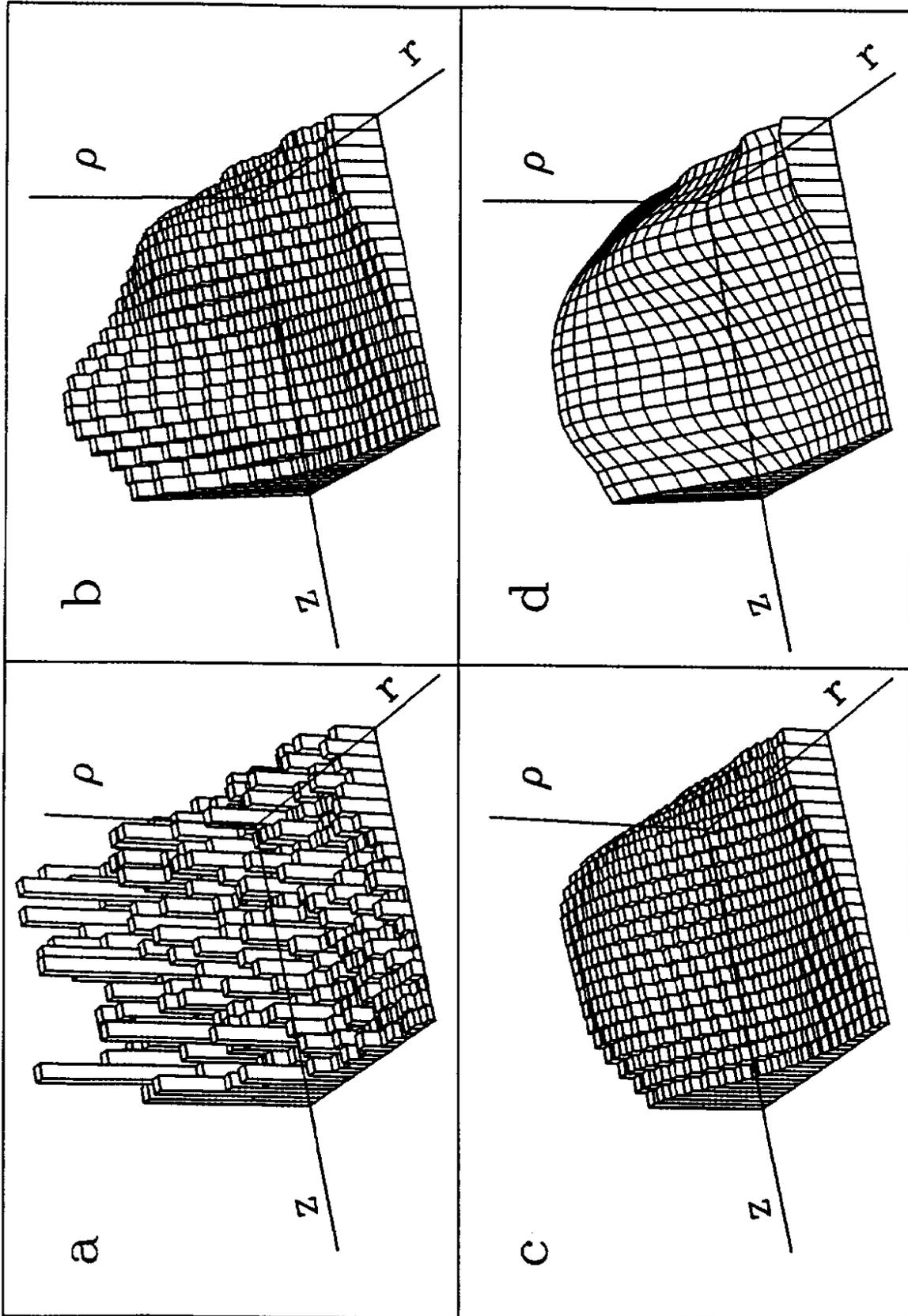


Fig. 9 Test of algorithm on (a) histogram based on eq. 32 with relative error $\sigma = 0.50$. Predictions of algorithm with boundary condition eq. 19 applied using (b) extra points at $r < 0$, (c) symmetry with respect to $r = 0$, (d) linear interpolation on equispaced grid and symmetry with respect to $r = 0$. The ordinate in (a) is compressed by a factor of 0.87 to accommodate fluctuations.

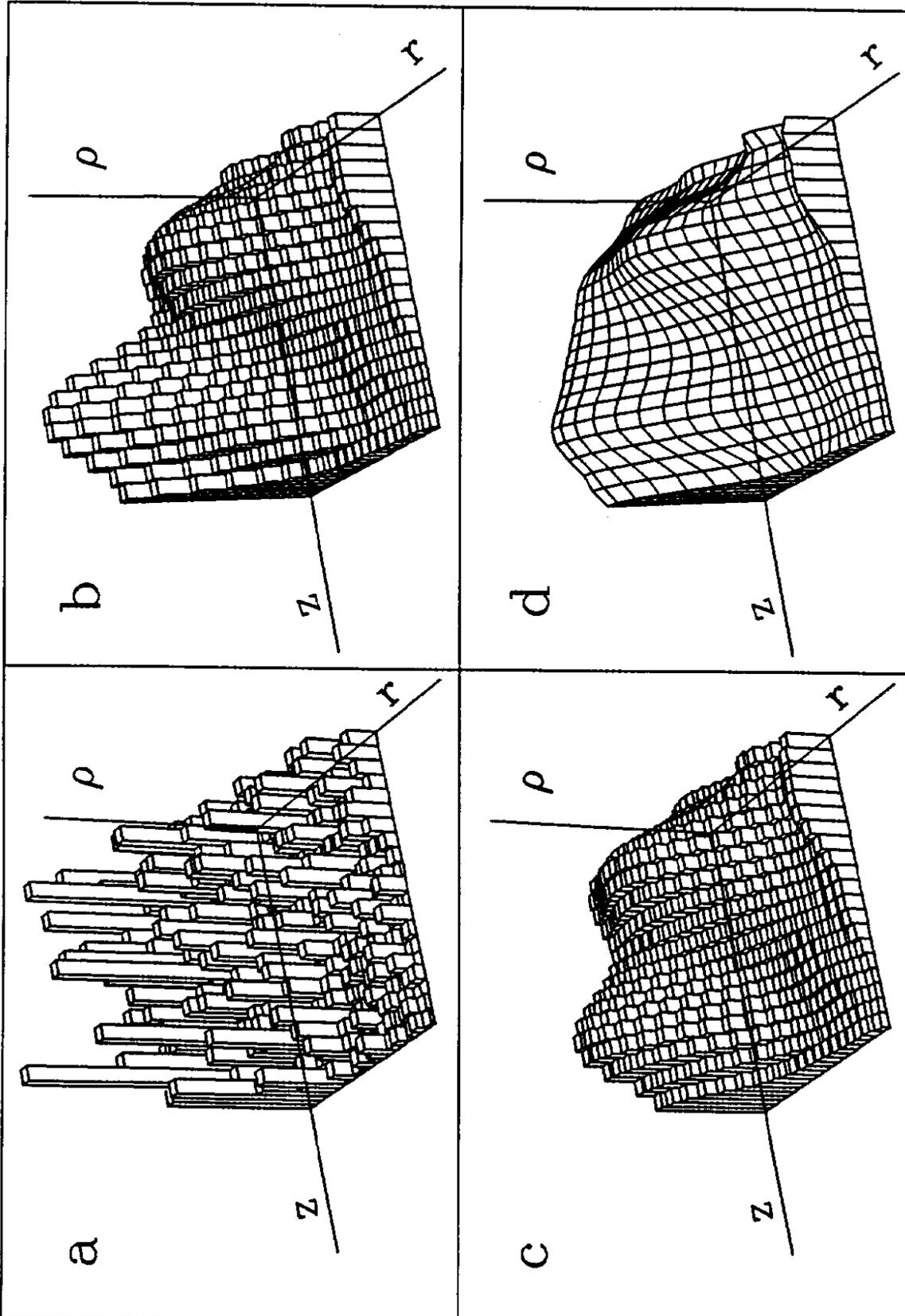


Fig. 10 Test of algorithm on (a) histogram based on eq. 32 with relative error $\sigma = 1.0$. Predictions of algorithm with boundary condition eq. 19 applied using (b) extra points at $r < 0$, (c) symmetry with respect to $r = 0$, (d) linear interpolation on equispaced grid and symmetry with respect to $r = 0$. The ordinate in (a) is compressed by a factor of 0.62 to accommodate fluctuations.

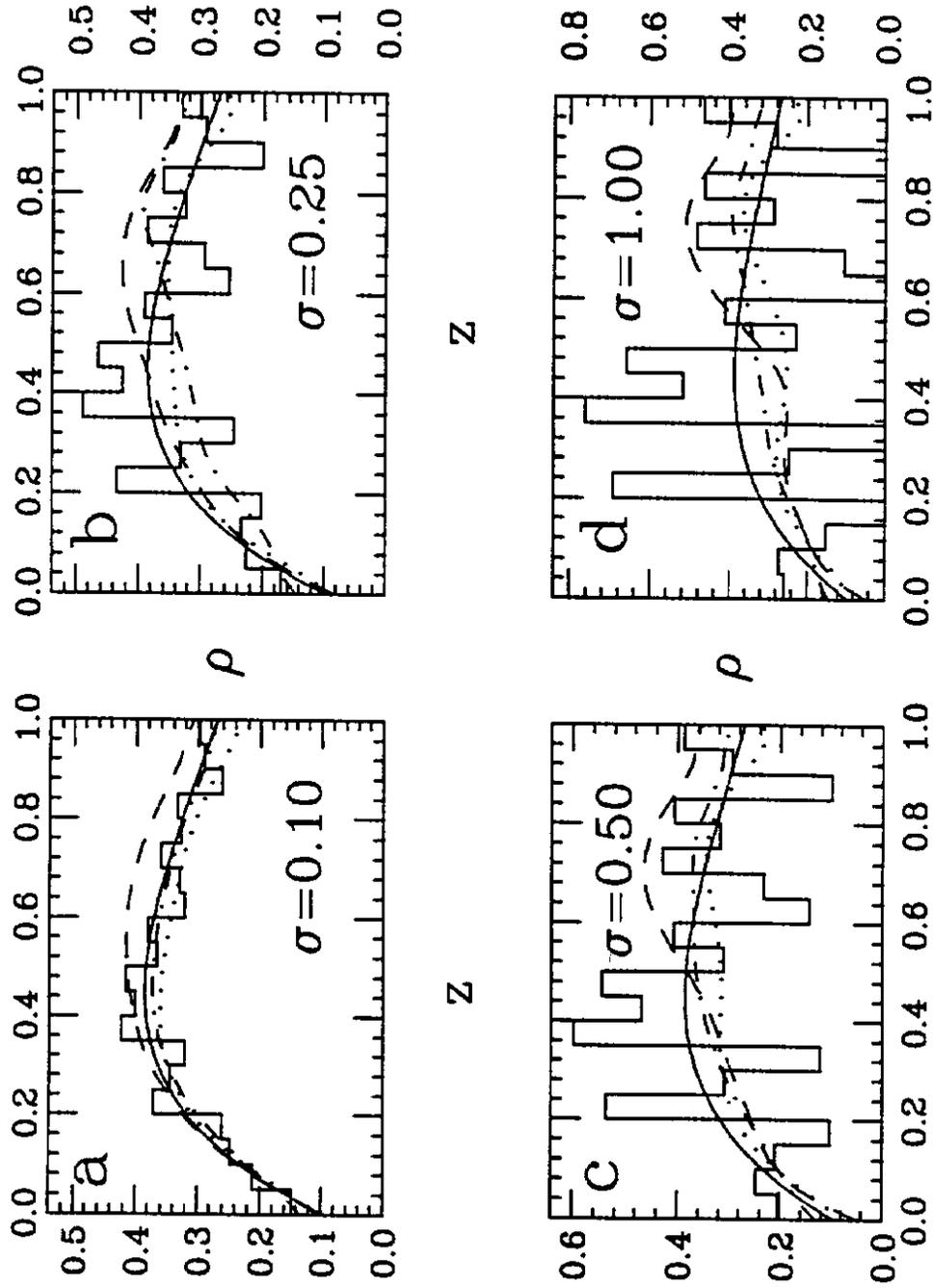


Fig. 11 Test of algorithm as in figs. 7-10 at $r = 0$. Solid line is eq. 32, histogram is part of two dimensional histogram closest to $r = 0$, dashed curve corresponds to method (b) of figs. 7-10, dotted curve to (c), and dash-dot curve to (d).

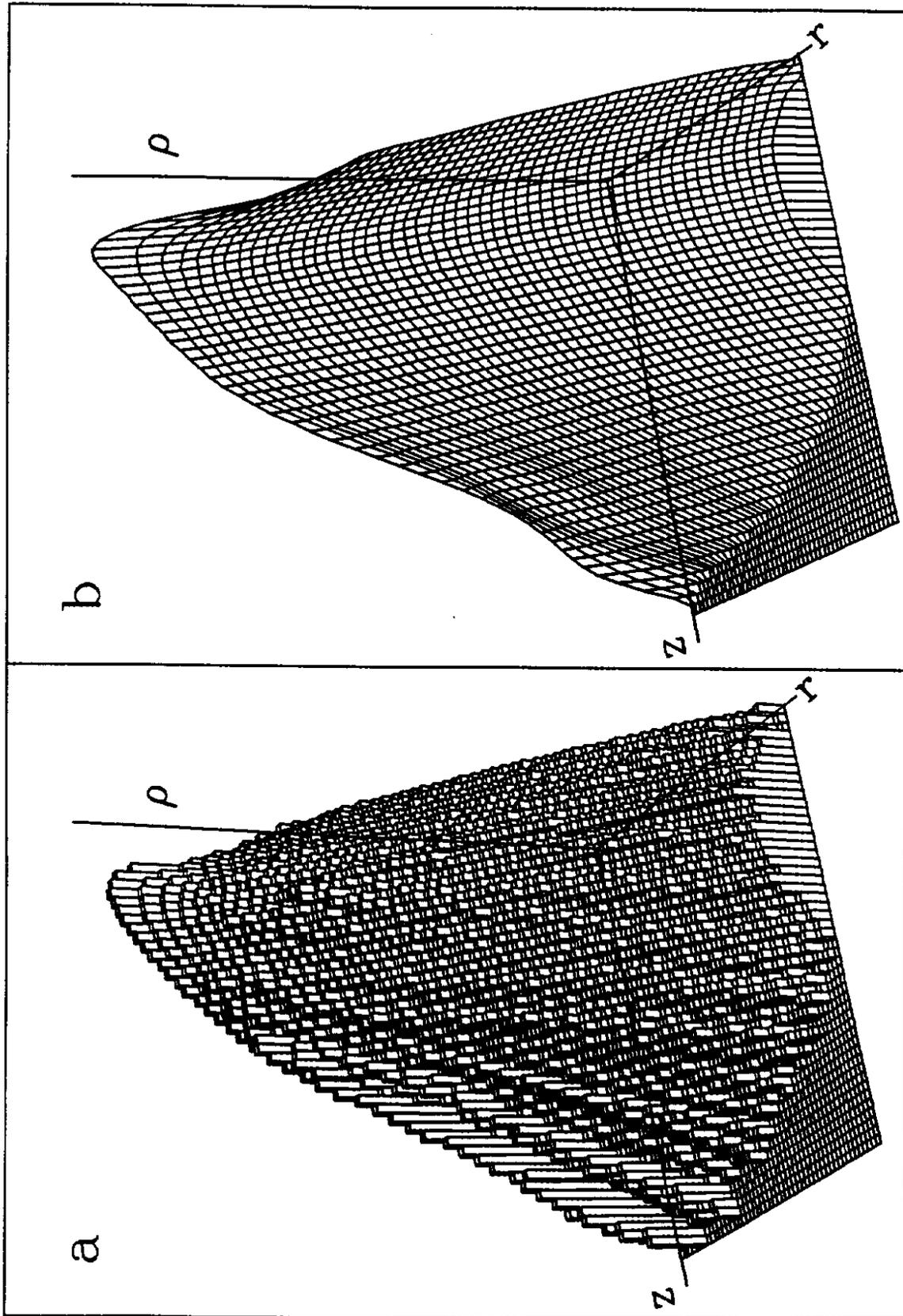


Fig. 12 (a) Two dimensional histogram of realistic CASIM dose calculation and (b) smooth approximation using algorithm (see text). Vertical scale is logarithmic and extends over about twenty orders of magnitude.

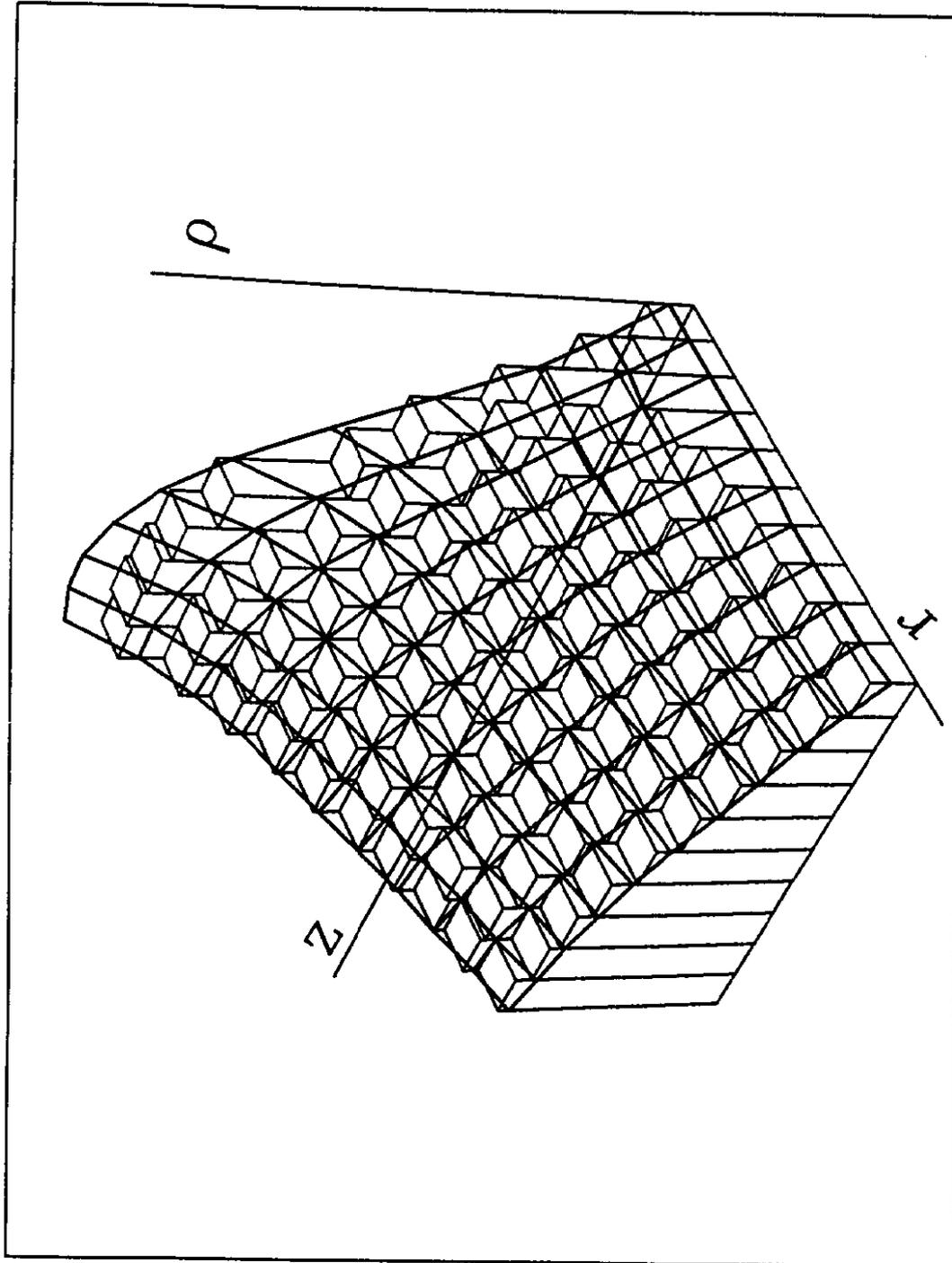


Fig. 13 Enlarged portion, at small r and small z , of fig. 12 with smooth approximat superimposed on histogram.