Approximate Representation of Functions of Several Variables in Terms of Functions of One Variable

H. L. Frisch and C. Borzi
Chemistry Department, State University of New York at Albany, Albany, New York 12222

G. Ord
Applied Mathematics Department, University of Western Ontario, London, Ontario, Canada

J. K. Percus
Courant Institute and Physics Department, New York University, New York, New York 10012

G. O. Williams
Mathematics Department, Manhattan College, New York, New York 10471
(Received 15 February 1989)

We present a procedure of practical use for representing functions of several variables as superpositions of functions of only one variable. We show how the procedure works when applied, for example, to the location of global minima. Our numerical examples are restricted here, for simplicity, to functions of two variables. The straightforward extension to functions of more variables will be discussed elsewhere.

PACS numbers: 02.30.+g

In diverse physical contexts ranging from many-body problems to coding of image data and conformational analysis of polyatomic molecules, the simultaneous variation of several variables obscures easy access to practical solutions. We seek to represent, to some degree of accuracy, functions of several variables as finite superpositions of functions of a single variable. We show here a practical implementation of such an approximation which could clarify some of the problems mentioned above. Details of our work, as well as further concrete applications, will be published elsewhere.

Following Arnol'd[1] and Kolmogorov's[2] solution of Hilbert's thirteenth problem,[3] using the more realizable version presented by Lorentz[4] we know that given a continuous function of s variables $f=f(x_1,\ldots,x_3)$ defined on the unit hypercube,

$$f\in C[\mathbb{R}^s],\quad f:[0,1]^s\rightarrow\mathbb{R},$$

there exists a continuous function of a single argument $g(z):[0,s]\rightarrow\mathbb{R}$ such that

$$f(x_1,\ldots,x_s)=\sum_{q=0}^{2s} g(z_q), \tag{1}$$

where the argument z_q is an embedding of the originally bounded multidimensional domain onto a one-dimensional bounded interval $z_q:[0,1]^s\rightarrow[0,s]$ defined as

$$z_q=\sum_{j=1}^{s}\lambda_j\phi_q(x_j). \tag{2}$$

Here the λ_i are rationally dependent numbers and $\phi_q(x)$ are monotonically increasing functions defined on special self-scaled partitions $[0,1]$. These ϕ's can be iteratively constructed in a computer following the proof of the lemma provided by Lorentz on p. 170 of Ref. 4. The partitions of the domain, conveniently refined, are shifted among themselves for different values of the index q, rendering a multiple $(2s+1)$-fold coverage of the domain on which the ϕ's are defined (we are using three refinements of these partitions in the example below). The whole process consists of the iteration of Eq. (1) on the error committed at the previous step and its details can be found on p. 173 in Ref. 4. The $g(z)$ is uniquely defined at each iteration of Eq. (1). It is constant on the disjoint segments of the z axis which are images of the elements of the partitions of the domain [through Eq. (2)] and is continued linearly among those segments. The global aspect of $g(z)$ is that shown in Fig. 2, where the local structure is hidden due to the coarse scale on z. That local structure can be easily recovered by blowing up the scale on the z axis.

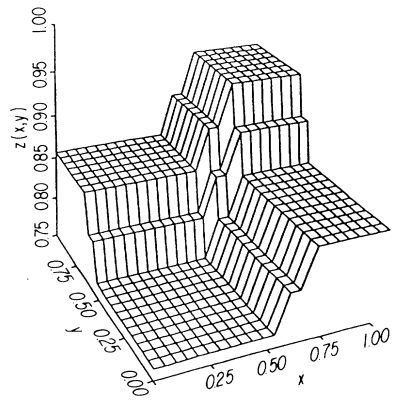

FIG. 1. Embedding of the unit square given by $z_2=\phi_2(r)+\lambda\phi_2(y)$ with ϕ_2 as defined in Ref. 4 and $\lambda=0.912\,354\,278\,387$.

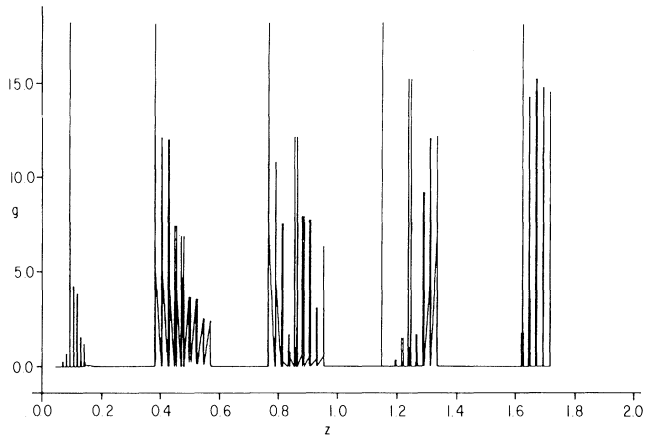

FIG. 2. Approximating function $g(z)$ for the function in Fig. 3, corresponding to the first iteration of Eq. (1).

The principal point we make in this Letter is that, at least for functions of two variables for which we have already implemented the procedure, the approximation works surprisingly well with a few iterations of Eq. (1). The extension to more variables is straightforward. The criteria of goodness we are using are (1) the overall geometrical appearance of the reconstructed function, (2) the relative norm, θ_k, of the error committed at the kth step, and (3) the local shifting of the extrema.

To clarify our observations, we illustrate with a graphical numerical example for $s=2$ in a number of figures below: Figure 1 shows, qualitatively, the appearance of the embedding (2). In Fig. 2 we show a typical approximating function $g(z)$ (the one which corresponds to the first iteration in Fig. 3). In Fig. 3 we present the geometrical appearance of an arbitrarily chosen function and the reconstructed pictures after one and five iterations of the procedure. We input the analytical expression for a superposition of truncated Gaussians. In general, it is enough to input a numerical table of values (which could be measured data points) provided it is extensive: The values at the centers of the elements of the $(2s+1)$ shifted partitions have to be provided.

We note the following:

(1) Convergence of this method of approximation is amazingly quick, as can be seen in Fig. 3. (The norms of the errors are given in Table I.) That makes the procedure of practical interest.

(2) We stress that the information on the "target" function is contained in the g's. The ϕ's and λ's are universal and, although the computation of the ϕ's is tedious and time consuming, it has to be done only once.

(3) The rank to which the partition of the domain has to be refined is determined by the rate of oscillation (see Ref. 4) of the target function on the smallest squares. Three refinements suffice for the functions we have examined.

(4) The outcome of the approximate representation is

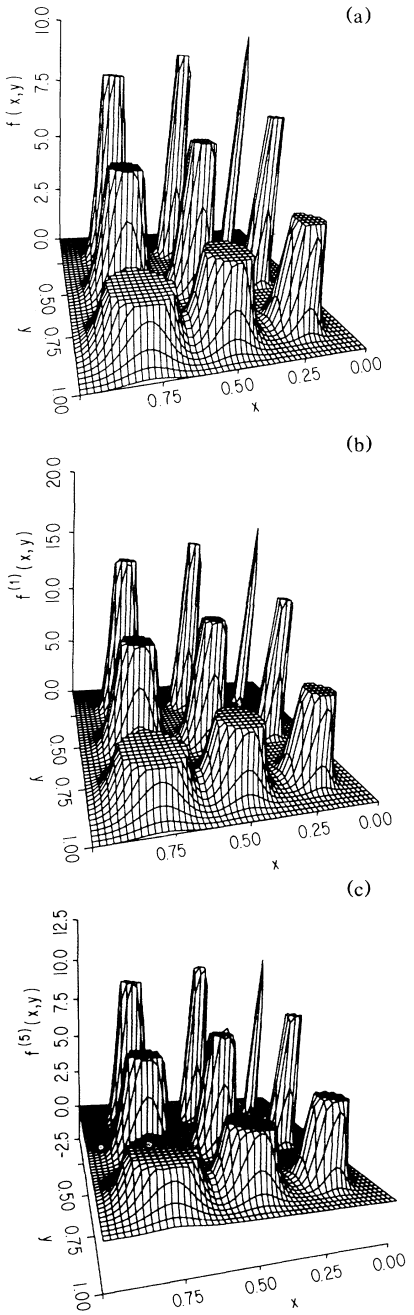

FIG. 3. Overall geometrical view of (a) a sum of truncated Gaussians $f(x,y)$ and (b) its first $f^{(1)}(z,y)$, and (c) fifth $f^{(5)}(x,y)$ representations through iterations of Eq. (1). The values of the relative norms of the errors are $\theta_1=0.669$ in (b) and $\theta_5=0.1336$ in (c).

a uniform magnification (which is then reduced and adjusted through sequential iterations) on top of which a local shifting occurs. This makes the approach, already at its first step, especially suited for finding global extrema among multiple ones.

To locate immediately the global extremum of a func-

TABLE I. Location of global extrema for the function in Fig. 3, and the one given by Eq. (3) as obtained from the first iteration of the procedure.

		Exact	From $g(z)$	ϵ
Global maximum for the function in Fig. 3	$x_{\max}$	0.2	0.193	0.035
	$x_{\max}$	0.2	0.193	0.035
Global minimum for the function in Eq. (3)	$x_{\min}$	0.0	0.01	0.01
	$x_{\min}$	0.0	0.01	0.01

tion of several variables (the maximum in the two-dimensional example below) one need only examine the approximating $g(z)$ at the first iteration of Eq. (1). The shape of the $g(z)$ will be similar to the one depicted in Fig. 2 with a number of cumulations of lines equal to $2s+1$ for s variables. One finds the global maximum of $g(z)$ at each cumulation at, say, $z_g^{\max}$ by ocular inspection (it is detected as the highest line). The scale on the z axis must be conveniently expanded around those maxima. One then obtains the coordinates in the original space via the inverse of Eq. (2). This locates the maximum up to an error of the order of the refinement of the working partition. Any other local procedure could increase the accuracy. To illustrate this we have obtained the location of the maximum from Fig. 2 as the average of $z_0^{\max}$ to $z_4^{\max}$ by the inverse of Eq. (2). These are compared with the exact values in Table I where ϵ is the relative error.

We treat the black-white intensity of image data as a continuous function of two variables defined constant over each pixel (that is, a superposition of Heaviside functions). The encoding of this information as a one-dimensional function can be done via $g(z)$ from Eq. (1). This also makes it easier to identify and recognize patterns.

We tested the global-extremum-finder method through the single-variable representation in another two-variable example which has been recently studied,[5,6]

$$f(x,y)=ax^2+by^2+c\cos(\gamma x)+d\cos(\delta y)\,, \qquad (3)$$

$$a,b>0,\quad c,d<0\,.$$

The representation is not essential here because (3) is already a sum of functions of one variable. Nevertheless, our method locates the global minimum in a few seconds of CPU time on a VAX 8650, as we show in Table I.

[1]V. I. Arnol'd, Dokl. Akad. Nauk SSSR **108**, 179 (1956).
[2]A. Kolmogorov, Dokl. Akad. Nauk. SSSR **114**, 679 (1957).
[3]D. Hilbert, Bull. Am. Math. Soc. **8**, 461 (1902).
[4]G. G. Lorentz, *Representation of Functions* (Chelsea, New York, 1986).
[5]I. O. Bohachevsky, M. E. Johnson, and M. L. Stein, Technometrics **28**, 209 (1986).
[6]L. Piela, I. Kostrowicki, and H. Scheraga (to be published).