On Smoothing Three-Dimensional Monte Carlo Ion Implantation Simulation Results

Clemens Heitzinger, Andreas Hössinger, and Siegfried Selberherr, *Fellow, IEEE*

***Abstract*—An algorithm for smoothing results of three-dimensional (3-D) Monte Carlo ion implantation simulations and translating them from the grid used for the Monte Carlo simulation to an arbitrary unstructured 3-D grid is presented. This algorithm is important for joining various process simulation steps, where data have to be smoothed or transferred from one grid to another. Furthermore, it is important for integrating the ion implantation simulator into a process flow. One reason for using different grids is that for certain Monte Carlo simulation methods, using orthogrids is mandatory because of performance reasons.**

The algorithm presented sweeps a small rectangular grid over the points of the new tetrahedral grid and uses approximation by generalized Bernstein polynomials. This approach was put on a mathematically sound basis by proving several properties of these polynomials. It does not suffer from the adverse effects of least squares fits of polynomials of fixed degree as known from the response surface method.

The most important properties of Bernstein polynomials generalized to cuboid domains are presented, including uniform convergence, an asymptotic formula, and the variation diminishing property. The smoothing algorithm which works very fast is described and, in order to show its applicability, the resulting values of a 3-D real world implantation example are given and compared with those of a least squares fit of a multivariate polynomial of degree two, which yielded unusable results.

***Index Terms*—Approximation methods, integrated circuit ion implantation, Monte Carlo methods, polynomials.**

I. Introduction

After a Monte Carlo simulation of ion implantation on an orthogrid, the question arises how to translate the resulting values, i.e., concentrations, to an unstructured grid. In the Monte Carlo simulation, an orthogrid is commonly used in order to achieve workable simulation times, since calculating point locations, i.e., tracing the position of ions, dominates performance. For other subsequent simulations via, e.g., the finite element method, it is mandatory to use different, unstructured grids. Furthermore, the resulting values have to be smoothed in order to provide suitable input for the simulation of subsequent process steps like diffusion.

Thus, an algorithm for smoothing Monte Carlo ion implantation results has to meet the following demands.

- It has to work with unstructured target grids.
- It must provide suitable smoothing.
- Since the number of grid points in the target grid is usually large, it must not be computationally expensive.

One simple approach is to perform a least squares fit of a multivariate polynomial of a fixed degree, usually two, and to hope that this polynomial is a suitable approximation providing proper smoothing. This is known as the response surface methodology (RSM) [1] approach and has been used to a great extent in computer-aided design applications, but it often does not work satisfactorily. In order to solve this problem, generalizations of Bernstein polynomials were devised and their properties proven. Hence, a fast algorithm based on these polynomials was developed and applied to real world examples, where its advantages can be seen. Before describing this method, we will have a closer look at the RSM approach. The RSM approach will be compared with the proposed algorithm since least squares fits are a popular method.

The RSM approach can be summarized as follows. Let f be a continuous function on a multidimensional interval $I := \{(x_1, \ldots, x_m) | a_k \leq x_k \leq b_k\}$. On the interval I, a rectangular grid with points x_k is chosen. The RSM approximation rsm of f is the multivariate polynomial of a degree two or higher fixed degree, which is determined by a least squares fit such that

$$\sum_k \left(f(x_k) - \mathrm{rsm}(x_k)\right)^2$$

is minimal. Computing the coefficients of rsm is a well-known procedure [1]. RSM has been used extensively in computer-aided design applications, e.g., in [2]–[13].

Although it can be argued that the RSM approximation is based on a truncated Taylor series expansion

$$f(\mathbf{r}+\mathbf{a}) = \sum_{k=0}^{\infty} \left(\frac{1}{k!} (\mathbf{a} \cdot \nabla_{\mathbf{r}'})^k f(\mathbf{r}') \right) \Big|_{\mathbf{r}'=\mathbf{r}}$$

for a multivariate function f, it is important to note that this is a local approximation and quite different from a least squares fit for several points. In the Taylor series expansion, convergence occurs when the number of terms and, thus, the degree of the polynomial increases, whereas in the RSM approach, the degree of the approximating polynomial is fixed to an arbitrary low value. Increasing the degree is possible of course, but the choice is still arbitrary and the number of coefficients and, thus, the number of points required for the least squares fit increases abundantly.

Furthermore, the RSM suffers from the fact that a polynomial of fixed degree cannot preserve the global properties of the original function: the set of all polynomials of a certain fixed maximal degree is not dense in $C(X)$, $X \subset \mathbb{R}^p$ compact. There

Manuscript received May 10, 2002; revised December 6, 2002. This work was supported in part by the Christian Doppler Forschungsgesellschaft, Vienna, Austria, and in part by the Austrian Program for Advanced Research and Technology (APART) from the Österreichische Akademie der Wissenschaften. This paper was recommended by Associate Editor Z. Yu.

The authors are with the Institute for Microelectronics, Technical University of Vienna, A-1040 Vienna, Austria (e-mail: Heitzinger@iue.tuwien.ac.at).

Digital Object Identifier 10.1109/TCAD.2003.814259

do not exist any rigorous statements about approximating or smoothing properties. Moreover, using more and more points for the least squares fit is not a remedy and generally does not improve the RSM polynomial, while the computational effort is increased. A simple example for this fact is the exponential function.

Although the RSM approach can be improved by transforming the variables before fitting the polynomials, it has to be known *a priori* which transformations are useful and should be considered. If this knowledge is available, it can of course be applied to other approximation approaches as well.

Finally, an advantage of the RSM approach is the simple structure of the approximations: it is easy to deal with polynomials of degree two. However, in the algorithm proposed in the following, no polynomials have to be constructed explicitly and the computational effort for doing least squares fits is eliminated as well.

In the following, the properties of generalized Bernstein polynomials are discussed and the algorithm is described in detail. Finally, the two approaches are compared by looking at the results of a three-dimensional (3-D) ion implantation example with about 80 000 grid points.

II. Properties of Bernstein Polynomials

The Weierstraß Approximation Theorem states that continuous functions on compact intervals can be arbitrarily well approximated by polynomials. One constructive way to obtain such polynomials are Bernstein polynomials which were first introduced by Sergei N. Bernstein in the univariate case. A generalization to multidimensional intervals and its properties is presented in this section. Generalizations to multidimensional simplices using barycentric coordinates and other properties of Bernstein polynomials can be found, e.g., in [14]–[20].

In order to keep the formulae simple, only functions defined on the multidimensional intervals $[0,1]\times\cdots\times[0,1]$, i.e., the unit cubes in $\mathbb{R}^N$, are considered. Using affine transformations, it is straightforward to apply the formulae and results to arbitrary intervals.

1) Definition (Multivariate Bernstein Polynomials): Let $n_1,\ldots,n_m \in \mathbb{N}$ and f be a function of m variables. The polynomials

$$B_{f,n_1,\ldots,n_m}(x_1,\ldots,x_m) := \sum_{\substack{0\le k_j\le n_j\\ j\in\{1,\ldots,m\}}} f\left(\frac{k_1}{n_1},\ldots,\frac{k_m}{n_m}\right) \cdot\prod_{j=1}^{m}\left(\binom{n_j}{k_j}x_j^{k_j}(1-x_j)^{n_j-k_j}\right)$$

are called the multivariate Bernstein polynomials of f.

We note that $B_{f,n_1,\ldots,n_m}$ is a linear operator.

2) Theorem (Pointwise Convergence): Let $f:[0,1]^m\to\mathbb{R}$ be a continuous function. Then, the multivariate Bernstein polynomials $B_{f,n_1,\ldots,n_m}$ converge pointwise to f for $n_1,\ldots,n_m\to\infty$.

The property of pointwise convergence can be obtained from uniform convergence in the univariate case by going up in the number of dimensions one by one. Much more important, however, is uniform convergence.

3) Theorem (Uniform Convergence): Let $f:[0,1]^m\to\mathbb{R}$ be a continuous function. Then, the multivariate Bernstein polynomials $B_{f,n_1,\ldots,n_m}$ converge uniformly to f for $n_1,\ldots,n_m\to\infty$.

A reformulation of this fact is the following corollary.

4) Corollary: The set of all polynomials is dense in $C([0,1]^m)$.

By presupposing more knowledge about the rate of change of the function, namely a Lipschitz condition, an error bound is obtained.

5) Theorem: If $f: I:=[0,1]^m\to\mathbb{R}$ is a continuous function satisfying the Lipschitz condition

$$\|f(x)-f(y)\|_2 < L\,\|x-y\|_2$$

on I, then the inequality

$$\|B_{f,n_1,\ldots,n_m}(x)-f(x)\|_2 < \frac{L}{2}\left(\sum_{j=1}^{m}\frac{1}{n_j}\right)^{\frac{1}{2}}$$

holds.

6) Theorem (Asymptotic Formula): Let $I:=[0,1]^m$, let $f: I\to\mathbb{R}$ be a C^2 function, and let $x\in I$, then

$$\lim_{n\to\infty} n\big(B_{f,n,\ldots,n}(x)-f(x)\big) = \sum_{j=1}^{m}\frac{x_j(1-x_j)}{2}\frac{\partial^2 f(x)}{\partial x_j^2} \le \frac{1}{8}\sum_{j=1}^{m}\frac{\partial^2 f(x)}{\partial x_j^2}.$$

This asymptotic formula gives us information about the rate of convergence, and states that it depends only on the partial derivatives $\partial^2 f(x)/\partial x_j^2$. This is noteworthy, since it is often the case that the smoother a function is and the more that is known about its higher derivatives, the more properties can be proven, but in this case, only the second order derivatives play a role.

7) Theorem (Total Variation): Let $V(f,[a,b])$ be the total variation of f over $[a,b]$ and let $f:[0,1]\to\mathbb{R}$ be a continuous function. Then

$$V\left(B_{f,n},[0,1]\right)\le V\left(f,[0,1]\right)$$

where the equality sign holds if and only if the function f is monotone.

This means the approximation is smoother than the original function concerning the amount of total variation.

8) Theorem (Variation Diminishing Property): Let $Z(f,(a,b))$ be the number of real zeros of f in the interval (a, b) and let $f:[0,1]\to\mathbb{R}$ be a continuous function. Then

$$Z\big(B_{f,n},(0,1)\big)\le v(f)$$

where $v(f)$ is the number of changes of sign of f in [0, 1].

This last theorem is the reason for the excellent smoothing properties of polynomials of Bernstein type. It states that Bernstein polynomials should be used whenever a polynomial approximation is needed which does not oscillate more often about any straight line than the function to be approximated.

Concerning the numerical aspect, an implementation for univariate Bernstein polynomials was presented in [21]. The higher the degree of the approximation polynomial, the more care that

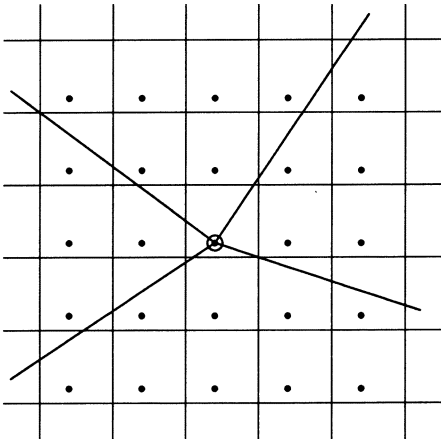

Fig. 1. In this figure, the calculations performed for one point of the unstructured target grid B are outlined in a two-dimensional example. They are analogous in higher dimensions. The thin orthogonal lines confine the cells of the initial grid A, the four sloped lines denote the unstructured grid, and the point in the middle is the one currently considered. The 5^2 points show which values are used for determining the approximating polynomial.

has to be taken in their numerical evaluation. In the cases needed for our applications, this is not an issue.

III. Algorithm

The algorithm works by constructing approximating multivariate Bernstein polynomials in the neighborhood of the points of the unstructured new grid. Let A be the initial isotropic homogeneous grid, where values are associated with the volume cells, as is usually the case in Monte Carlo simulations of ion implantations, and B an arbitrary grid where values are associated with the grid points. This grid is to be used in following simulations, and hence, it is determined by their demands. It is often an anisotropic inhomogeneous one.

For each point of grid B, m^d neighboring points are used for constructing an approximation value for the point considered (cf. Fig. 1), where $m \geq 3$, m odd, and d is the dimension. $m = 5$ was chosen in the example below and provides good smoothing results. At the boundary, the values of grid A are extended constantly. Thus, m^d points are used for constructing a multivariate Bernstein polynomial which is evaluated at the point in the middle in question. Note that it is not necessary to calculate the polynomial explicitly, since each polynomial is later evaluated at one point only. Additionally, it is not necessary to use an affine transformation by assuming that the convex hull of the neighboring points is $[0,1]^d$ and the middle point has coordinates $(1/2,\ldots,1/2)$.

Thus, for three dimensions and setting $n := m-1$, the values of the points of grid B are

$$B_{f,n,n,n}\left(\frac{1}{2},\frac{1}{2},\frac{1}{2}\right)=\frac{1}{8^n}\sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n}f_{k_1,k_2,k_3}\binom{n}{k_1}\binom{n}{k_2}\binom{n}{k_3}$$

where f_{k_1,k_2,k_3} are the values of the corresponding cell of grid A and $f_{0,0,0}$ has coordinates (0, 0, 0) and $f_{n,n,n}$ has coordinates (1, 1, 1).

One of the benefits of this algorithm is that it can be implemented in a straightforward manner in languages like C and Fortran using the expression for $B_{f,n,n,n}(1/2,1/2,1/2)$ given above. In order to minimize computation time, the values of the binomial coefficients can be precalculated and stored in arrays.

Furthermore, it is fast so that it can be used for grids containing hundreds of thousands of points. Due to the theorems given above, its smoothing and approximating properties are outstanding. Thus, it is faster, easier to implement, and approximates and smooths better than the RSM approach of fitting polynomials of fixed degree.

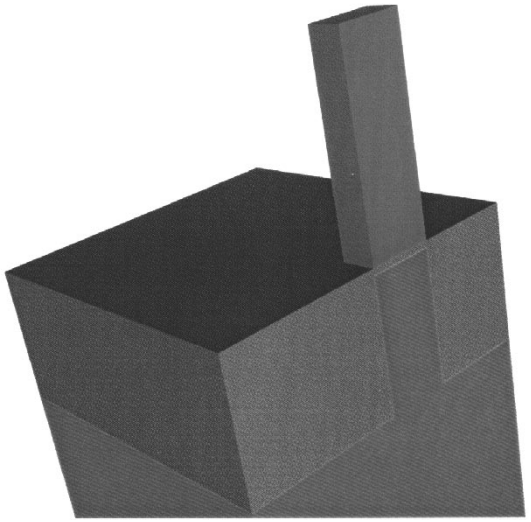

Fig. 2. View of the sample device, where the upper part consists of polysilicon, the middle part of silicon dioxide, and the lower part of silicon.

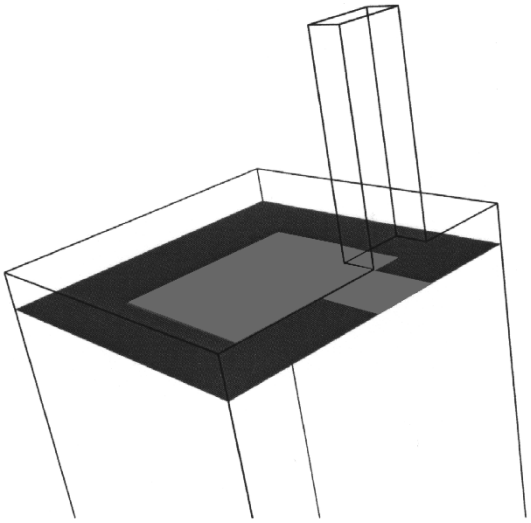

Fig. 3. Cut through the sample device, where the middle part is silicon and the outer part silicon dioxide.

IV. 3-D Example

The example is a 3-D CMOS structure as shown in Figs. 2 and 3, which consists of poly-silicon in the upper part, silicon dioxide in the middle part, and silicon in the lower part. A boron dose of 10^{13} cm^{-2} with an energy of 15 keV was implanted in a Monte Carlo simulation [22]–[24] using an isotropic homogeneous grid. The resulting concentration of boron interstitial atoms in [cm^{-3}] is shown in Figs. 4–8. The new anisotropic inhomogeneous grid with 78 651 grid points was generated by

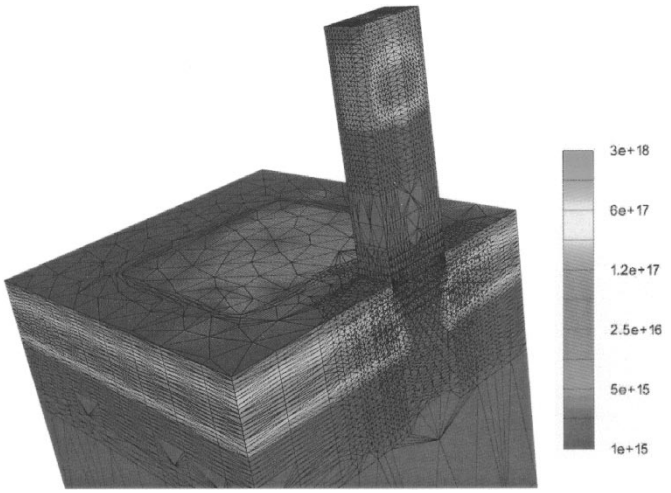

Fig. 4. Front view of the sample Monte Carlo ion implantation after smoothing using the new algorithm. The unstructured destination grid with 78 651 points is shown as well.

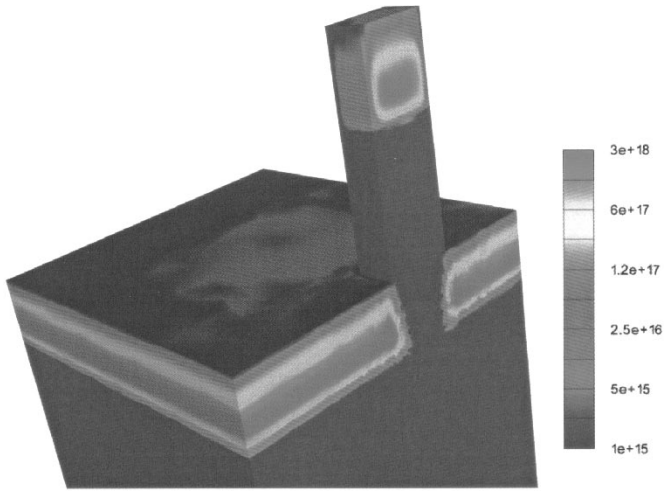

Fig. 5. Front view of the sample Monte Carlo implantation after smoothing using the new algorithm.

DELINK [25] and is additionally shown in Fig. 4. In Figs. 4, 5, 7, and 8, the new algorithm was applied on $5 \cdot 5$ grids, whereas in Fig. 6, least squares fits of polynomials of degree two on grids of the same size were performed.

Obviously, the result in Fig. 6 is inferior to the result yielded by the algorithm described in the previous section. In order to interpret the failure of the RSM method, it is important to note that the shape of the RSM polynomials of degree 2 does not allow enough change to adapt to the points to be approximated. Because of the inherent noise in the Monte Carlo simulation result, the shapes of the RSM approximations vary strongly between neighboring elements. Furthermore, because of the limited choice of approximating polynomials, the noise may even be amplified.

The new algorithm provides very good smoothing and yields concentration values at the grid points that can serve as input to subsequent simulation steps without problems. In this example the computation time on an Intel Pentium III processor at 1 GHz is 2.417 ms per point using the RSM method and 0.858 ms per point using the new algorithm.

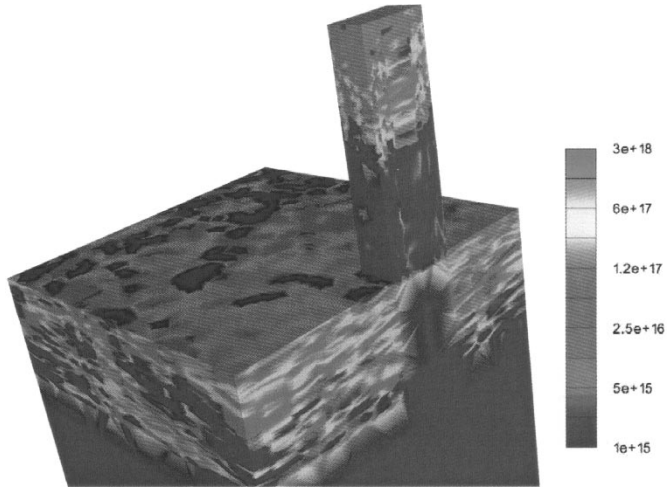

Fig. 6. Front view of the sample Monte Carlo ion implantation after extracting values using least squares fits of multivariate polynomials of degree two.

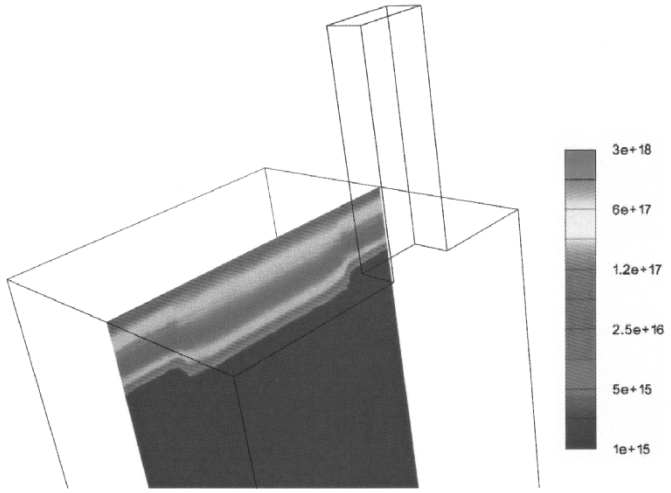

Fig. 7. Cut parallel to the front side of the sample Monte Carlo ion implantation after smoothing using the new algorithm.

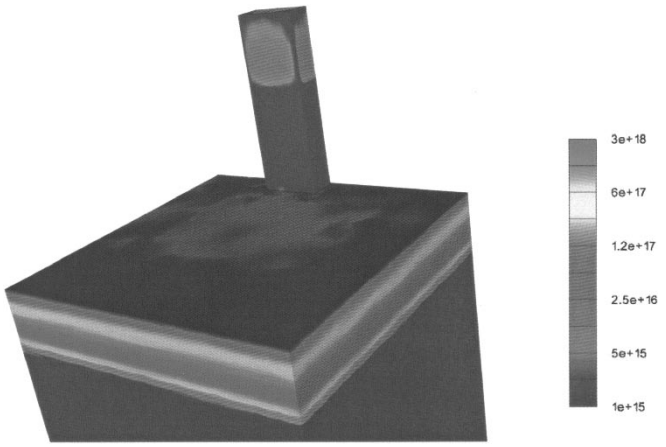

Fig. 8. Back view of the sample Monte Carlo ion implantation after smoothing using the new algorithm.

V. Conclusion

In summary, the properties of polynomials of fixed degree arising from least square fits were compared to those of general-

ized Bernstein polynomials on multidimensional intervals. The properties of the Bernstein polynomials were proven and presented, and it was found that these fulfill the requirements for approximations needed for smoothing Monte Carlo simulation results and translating them from ion implantation orthogrids to unstructured grids.

The polynomials and the algorithm devised provide the following benefits. First, they converge uniformly when the number of base points goes to infinity. Second, an asymptotic formula gives information about their rate of convergence. Third, total variation is decreased and the approximations do not oscillate more often about any straight line than the original function. This assures suitable smoothing. Fourth, the algorithm works very fast and is easy to implement using the specialized formula given, since the calculation of the actual approximating polynomials is avoided.

Finally, the new algorithm and its RSM counterpart were compared in a real world Monte Carlo ion implantation example, and the new algorithm was found to yield superior results, which can immediately be used for further simulations.

REFERENCES

[1] G. Box and N. Draper, *Empirical Model-Building and Response Surfaces*. New York: Wiley, 1987.

[2] J. Cornell, *How to Apply Response Surface Methodology*, 2nd ed. Milwaukee, WI: Amer. Soc. Quality Control, 1990.

[3] G. Gaston and A. Walton, "The integration of simulation and response surface methodology for the optimization of IC processes," *IEEE Trans. Semiconduct. Manufact.*, vol. 7, pp. 22–33, Feb. 1994.

[4] D. Boning and P. Mozumder, "DOE/Opt: a system for design of experiments, response surface modeling, and optimization using process and device simulation," *IEEE Trans. Semiconduct. Manufact.*, vol. 7, pp. 233–244, May 1994.

[5] R. Cartuyvels, R. Booth, S. Kubicek, L. Dupas, and K. De Meyer, "A powerful TCAD system including advanced RSM techniques for various engineering optimization problems," in *Simulation of Semiconductor Devices and Processes*, S. Selberherr, H. Stippel, and E. Strasser, Eds. Vienna, Austria: Springer, 1993, vol. 5, pp. 29–32.

[6] M. Welten, R. Clancy, M. Murphy, and W. Lane, "Total DFM approach for a 0.6 μm CMOS process," in *Proc. StatMetWS'96 [27]*, pp. 1–5.

[7] H. Stippel, P. Lindorfer, and J. Weintraub, "Response surface method for statistical process and device simulation," in *StatMetWS'96 [27]*, pp. 6–9.

[8] T. Waring, A. Walton, S. Ferguson, and D. Sprevak, "Application of covariance structures to improve the fit of response surfaces to simulation data," *IEEE Trans. Semiconduct. Manufact.*, vol. 12, pp. 366–374, Aug. 1999.

[9] V. Senez, T. Hoffmann, and A. Tixier, "Calibration of a two-dimensional numerical model for the optimization of LOCOS-type isolations by response surface methodology," *IEEE Trans. Semiconduct. Manufact.*, vol. 13, pp. 416–426, Nov. 2000.

[10] A. Alvarez, B. Abdi, D. Young, H. Weed, J. Teplik, and E. Herald, "Application of statistical design and response surface methods to computer-aided VLSI device design," *IEEE Trans. Computer-Aided Design*, vol. 7, pp. 272–288, Feb. 1988.

[11] W. Kanert, N. Krischke, and K. Wiesinger, "Optimization of DMOS transistors for smart power technologies by simulation and response surface methods," in *Simulation of Semiconductor Devices and Processes*, S. Selberherr, H. Stippel, and E. Strasser, Eds. Vienna, Austria: Springer, 1993, vol. 5, pp. 213–216.

[12] H. Masuda, F. Otsuka, Y. Aoki, S. Sato, and S. Shimada, "Response surface methods for submicron MOSFET's characterization with variable transformation technology," *IEICE Trans.*, vol. E 74, no. 6, pp. 1621–1633, 1991.

[13] H. Sato, K. Tsuneno, K. Aoyama, T. Nakamura, H. Kunitomo, and H. Masuda, "A new hierarchical RSM for TCAD-based device design to predict CMOS development," in *Proc. IEEE Int. Conf. Microelectron. Test Structures*, vol. 8, Mar. 1995, pp. 299–302.

[14] G. Lorentz, *Bernstein Polynomials*, 2nd ed. New York: Chelsea, 1986.

[15] I. Berezin and N. Zhidkov, *Computing Methods*. New York: Pergamon, 1965, vol. 1.

[16] I. Schoenberg, "On variation diminishing approximation methods," in *On Numerical Approximation*, R. Langer, Ed. Madison, WI: Univ. Wisconsin Press, 1959, pp. 249–274.

[17] R. Schnabl, "Eine Verallgemeinerung der Bernsteinpolynome," *Math. Ann.*, vol. 179, pp. 74–82, 1968.

[18] J. Adell, J. de la Cal, and M. S. Miguel, "On the property of monotonic convergence for multivariate Bernstein-type operators," *J. Approx. Theory*, vol. 80, no. 1, pp. 132–137, 1995.

[19] Y. Feng and J. Kozak, "Asymptotic expansion formula for Bernstein polynomials defined on a simplex," *Proc. Constructive Approx.*, vol. 8, pp. 49–58, 1992.

[20] T. Popoviciu, "Sur l'approximation des fonctions convexes d'ordre supérieur," *Mathematica*, vol. 10, pp. 49–54, 1935.

[21] Y.-F. Tsai and R. Farouki, "Algorithm 812: BPOLY: an object-oriented library of numerical algorithms for polynomials in Bernstein form," *ACM Trans. Math. Softw.*, vol. 27, no. 2, pp. 267–296, 2001.

[22] A. Hössinger, S. Selberherr, M. Kimura, I. Nomachi, and S. Kusanagi, "Three-dimensional Monte-Carlo ion implantation simulation for molecular ions," in *Proc. Electrochem. Soc.*, vol. 99-2, 1999, pp. 18–25.

[23] A. Hössinger, E. Langer, and S. Selberherr, "Parallelization of a Monte Carlo ion implantation simulator," *IEEE Trans. Computer-Aided Design*, vol. 19, pp. 560–567, May 2000.

[24] A. Hössinger, M. Radi, B. Scholz, T. Fahringer, E. Langer, and S. Selberherr, "Parallelization of a Monte-Carlo ion implantation simulator for three-dimensional crystalline structures," in *Proc. Simulation Semiconduct. Processes Devices*, Kyoto, Japan, Sept. 1999, pp. 103–106.

[25] P. Fleischmann, "Mesh Generation for Technology CAD in Three Dimensions," Ph.D. dissertation, Technical University of Vienna, Vienna, Austria, 1999.

[26] S. Selberherr, H. Stippel, and E. Strasser, Eds., *Simulation of Semiconductor Devices and Processes*. Vienna, Austria: Springer, 1993, vol. 5.

[27] *Proc. 1st Int. Workshop Statist. Metrol.*, Honolulu, HI, 1996.

Clemens Heitzinger was born in Linz, Austria, in 1974. He received the Diplom-Ingenieur degree (honors) in technical mathematics and the doctoral degree in technical sciences (honors) from the Technical University of Vienna, Vienna, Austria, in 1999 and 2002, respectively.

From March to May 2001, he was a Visiting Researcher at the Sony Technology Center, Hon-Atsugi, Tokyo, Japan. His scientific interests include process simulation for semiconductor devices and applied mathematics for simulation in microelectronics.

Andreas Hössinger was born in St. Pölten, Austria, in 1969. He received the Diplom-Ingenieur degree in technical physics, in 1996, and the doctorate degree, in 2000, from the Technical University of Vienna, Vienna, Austria, where he is currently enrolled as a postdoctoral researcher.

In 1998, he was a Visiting Researcher with Sony Technology Center, Atsugi, Japan. In 2001, he was a Visiting Researcher at LSI Logic, Santa Clara, CA, within the scope of a cooperate research project on 3-D process simulation. His research interests include process simulation with special emphasis on 3-D applications.

Siegfried Selberherr (M'79–SM'84–F'93) was born in Klosterneuburg, Austria, in 1955. He received the Diplom-Ingenieur degree in electrical engineering and the doctoral degree in technical sciences from the Technical University of Vienna, Vienna, Austria, in 1978 and 1981, respectively.

He has held the *venia docendi* on computer-aided design since 1984. Since 1988, he has been Head of the Institute for Microelectronics and, since 1999, has been Dean of the Faculty for Electrical and Information Technology. His current topics are modeling and simulation of problems for microelectronics engineering.