

Binary and Phase Shifting Mask Design for Optical Lithography

Yong Liu, *Student Member, IEEE*, and Avidesh Zakhor, *Member, IEEE*

Abstract—We propose a number of pre-distorted mask design techniques for binary and phase-shifting masks. Our approach is based on modeling the imaging mechanism of a stepper by the Hopkins equations and taking advantage of the contrast-enhancement characteristics of photoresist. Optimization techniques such as the branch and bound algorithm and simulated annealing algorithm are used to systematically design pre-distorted masks under incoherent and partially coherent illumination. Computer simulations are used to show that the intensity contour shapes and developed resist shapes of our designed mask patterns are sharper than those of conventional masks. The designed phase-shifting masks are shown to result in higher contrast as well as sharper contours than binary masks. An example of phase conflicting mask designed via our algorithm is shown to outperform a simple intuitive design. This example indicates that a fairly general design procedure consisting of alternating phase shifts and our optimized phase-shift mask is a viable candidate for future phase-shifting mask design.

I. INTRODUCTION

AS THE minimum feature size in VLSI circuits drops into submicron region, distortions due to optical diffraction can no longer be neglected. An established practice in microphotography to overcome such distortions is to introduce deliberate distortions in the mask artwork. The corrections are usually based on experience gained by a process of trial and error. For the recently developed phase shifting masks, experience is scarce, and intuitions are hard to develop. Therefore, a systematic technique to determine suboptimal, if not optimal, pre-distorted masks can facilitate conventional and phase-shifting mask design.

In this paper, we propose ways of designing pre-distorted masks by formulating it as a signal/image synthesis problem. Typical image synthesis problems [1] involve designing an input image to a system with known characteristics in such a way that the output resembles a pre-described image as much as possible. In practice, physi-

cal limitations of the systems under consideration result in a set of constraints on the class of images at the output of the system. For instance, in microlithography applications, it is not possible to generate patterns with arbitrarily sharp corners at the output of a stepper since the optical system is band-limited. To overcome this, our approach is to devise a set of specifications for the output image, so that the designed input masks satisfy the output specifications as much as possible. In doing so, we propose a set of specifications in Section III and two optimization based techniques in Sections IV and V.

Our first technique maximizes the sum of intensity slopes at the transition regions of a mask, subject to inequality constraints associated with non-transition areas. By transition regions of a mask, we mean areas in which the binary pattern includes a transition from one to zero or vice versa. Under incoherent illumination this criterion results in a linear binary programming problem with linear constraints, which can be solved via the well known branch and bound algorithm [2]. As we will see, this criterion results in sharper output intensity contours than those of the mean squared error criterion [3].

For partially coherent illumination, the above formulation results in a binary quadratic programming problem with quadratic constraints. The principle of branch and bound algorithm still applies, but the implementation is more involved than that of the incoherent case. To this end, we have chosen a simpler formulation in which we convert all the constraints into one objective function with proper weights. Specifically, we choose the total objective function to be a weighted sum of objective functions corresponding to different sampling points. The objective function at a particular sampling point is chosen to be a nonlinear function of the error at that point and simulated annealing is used to avoid local minima. Phase shifting masks designed by this formulation will be shown to result in sharper contours than those due to regular masks without reducing the contrast.

The outline of this paper is as follows. Section II includes system modeling, Section III is on design specifications, section IV is on the branch and bound approach and its application to binary mask design, Section V is on formulation and application of simulated annealing to phase shifting mask design, and Section VI includes discussions and conclusions.

Manuscript received May 24, 1991; revised November 26, 1991. This work was supported in part by the Semiconductor Research Corporation, Grant 92-DC-008, and in part by the National Science Foundation PYI Grant MIP-9057466. IBM, Kodak, TRW, and Joint Services Electronics Project. Y. Liu was partially supported by the K. C. Wong Education Foundation.

The authors are with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720. IEEE Log Number 9106754.

II. SYSTEM MODELING

The imaging mechanism of a stepper can be modeled by Hopkins equation [4]–[6]:

$$I(f, g) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} T(f' + f, g' + g; f', g') \cdot F(f' + f, g' + g) F^*(f', g') df' dg' \quad (1)$$

$$I(x, y) = \mathcal{F}^{-1}[I(f, g)]$$

where $I(f, g)$ is the Fourier transform of the output image intensity $I(x, y)$, $F(f, g)$ is the Fourier transform of mask transmission function $F(x, y)$, which is usually 1 in transparent areas and 0 in opaque areas of the mask, and $T(f', g'; f'', g'')$ is the Transmission Cross-Coefficient (TCC) of the optical system, which summarizes all the information about the imaging system and illumination. The TCC function is given by

$$T(f', g'; f'', g'') = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} J(f, g) K(f + f', g + g') \cdot K^*(f + f'', g + g'') df dg \quad (2)$$

where $J(f, g)$ is Fourier transform of mutual intensity of the light at the object and $K(f, g)$ is frequency response function of the imaging system. If circular pupils are used, then the frequency response function on focus plane is given by

$$K(f, g) = \begin{cases} 1 & f^2 + g^2 < \left(\frac{NA}{\lambda M}\right)^2 \\ 0 & f^2 + g^2 \geq \left(\frac{NA}{\lambda M}\right)^2 \end{cases} \quad (3)$$

where NA is numerical aperture, λ is wavelength of light source, and M is reduction factor of the lens system. Assuming incoherent light source, the mutual intensity function J is

$$J(f, g) = \begin{cases} \frac{\lambda^2}{\pi s^2 NA^2} & f^2 + g^2 < \left(\frac{sNA}{\lambda}\right)^2 \\ 0 & f^2 + g^2 \geq \left(\frac{sNA}{\lambda}\right)^2 \end{cases} \quad (4)$$

where s is the partial coherence factor given by

$$s = \frac{\text{NA of condenser}}{\text{NA of imaging optics}} \quad (5)$$

If we quantize the mask image into small square pixels, as shown in Fig. 9, with a_{ij} representing the transmission variable at the (i, j) th pixel taking on values $\{0, 1\}$, then the mask transmission function $F(x, y)$ can be expressed as

$$F(x, y) = \sum_{i=1}^N \sum_{j=1}^N a_{ij} \psi_{ij}(x, y) \quad (6)$$

where N is the number of pixels in each dimension and $\psi_{ij}(x, y)$ is a unit square pulse function located at the ij th pixel. Taking Fourier transform of (6) and substituting it into (1), we obtain the following expression for the output image intensity

$$I(x, y) = \sum_{i=1}^N \sum_{j=1}^N \sum_{l=1}^N \sum_{m=1}^N a_{ij} a_{lm} G_{ijlm}(x, y) \quad (7)$$

where $G_{ijlm}(x, y)$ is the intensity due to the interaction between the (i, j) th pixel and the (l, m) th pixel and is given by

$$G_{ijlm}(f, g) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} T(f' + f, g' + g; f', g') \cdot \psi_{ij}(f' + f, g' + g) \psi_{lm}^*(f', g') df' dg' \quad (8)$$

For $(i, j) = (l, m)$, $G_{ijlm}(x, y)$ is the intensity due to the (i, j) th pixel alone.

For incoherent illumination, the cross product terms are all zero, and the system becomes linear shift invariant. Under this condition, since $a_{ij}^2 = a_{ij}$ and $G_{ijij}(x, y)$ is simply a shifted version of $G_{0000}(x, y)$, we obtain an output intensity of the form

$$I(x, y) = \sum_{i=1}^N \sum_{j=1}^N a_{ij} g_r(x - id, y - jd) \quad (9)$$

where $g_r(x, y) = G_{0000}(x, y)$ is the output intensity due to a pixel located at the origin.

III. DESIGN SPECIFICATIONS

The design specifications are best derived by way of an example. Suppose the desired output image is a pattern of two infinitely long bars, shown in Fig. 1. The desired binary aerial image is zero in domain A and one in domain B or the complement of A . Let C and D represent contours surrounding the boundaries of regions A and B respectively, where contour C is in region A and contour D is in region B .

We have chosen the following constraints as our design specifications:

$$I(\vec{x}) \leq a \quad \vec{x} \in A \quad (10)$$

$$I(\vec{x}) \geq b \quad \vec{x} \in B \quad (11)$$

$$I(\vec{x}) \leq c \quad \vec{x} \in C \quad (12)$$

$$I(\vec{x}) \geq d \quad \vec{x} \in D \quad (13)$$

where \vec{x} represents spatial location (x, y) , and a, b, c and d are the corresponding intensity threshold values in regions A, B, C and D , respectively.

Since it is difficult to check the design specifications at all the points in A, B, C and D as shown in inequalities (10)–(13), we have chosen to discretize the regions and the contours. Fig. 2 shows the cross section of aerial im-

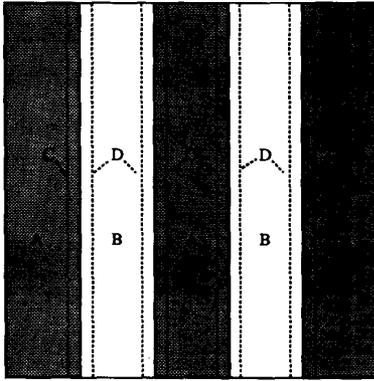


Fig. 1. Desired binary image of two infinitely long bars. The desired image is 1 in region B and 0 in region A. The regions between contours C and D are transition regions in the specifications.

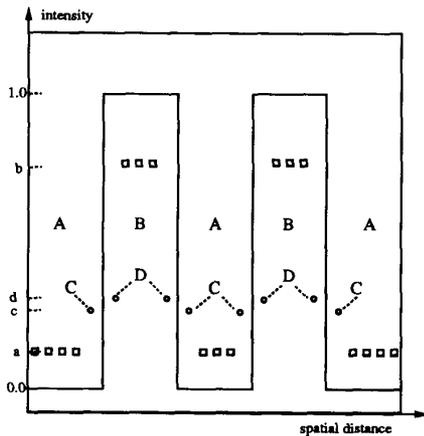


Fig. 2. Cross section diagram of Fig. 1 illustrating the specifications for the two infinitely long bars as a desired image. The square and circular dots are the sampling points enforcing the constraints.

age together with the sampling points used for enforcing the constraints of inequalities (10)–(13). Intuitively speaking, as long as the samples are dense enough, satisfying the constraints at the sampling points results in satisfying the constraints in an entire region. This is because the image intensity is a band-limited function. Fig. 4 shows the sampling point distribution for an elbow. To increase flexibility, we have chosen different densities for contour samples and domain samples.

Taking into account the process of photoresist development and assuming that the normalized 0.3 intensity contours correspond to the line edges after development, the intensity threshold values a , b , c and d can be used to represent a simple photoresist model, shown in Fig. 3, where $d = c = 0.3$. Our problem is then to find a set of pixel variables, configurations or states, such that the output intensity $I(x, y)$ satisfies all the constraints which describe the desired output image at an appropriate set of sampling points.

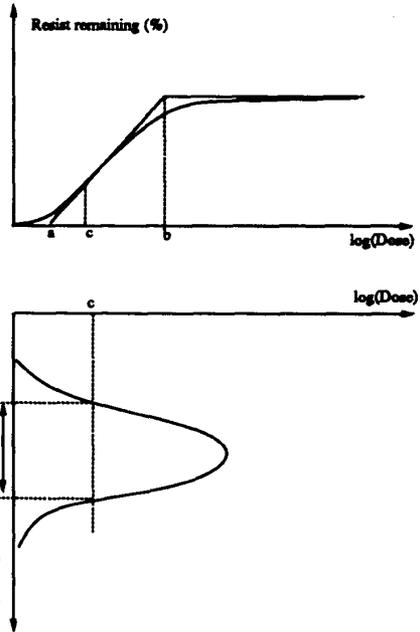


Fig. 3. A simple model of photoresist. As a rule of thumb, the line width corresponds to the 0.3 contour of light intensity when the light intensity is normalized by that of a large feature.

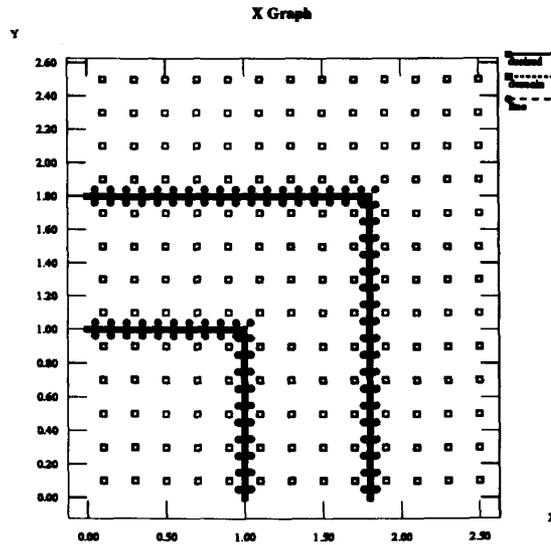


Fig. 4. Sampling point distribution for an elbow. The dark lines are the desired contours. The regions outside (between) the two dark lines are region A (B), and the sampling points are the white squares. The lines formed by the circular sampling points inside region A (B) corresponds to contour C (D).

IV. OPTIMIZATION BY BRANCH AND BOUND ALGORITHM FOR BINARY MASKS

We have chosen the objective function for our optimization problem to be the sum of intensity slopes across

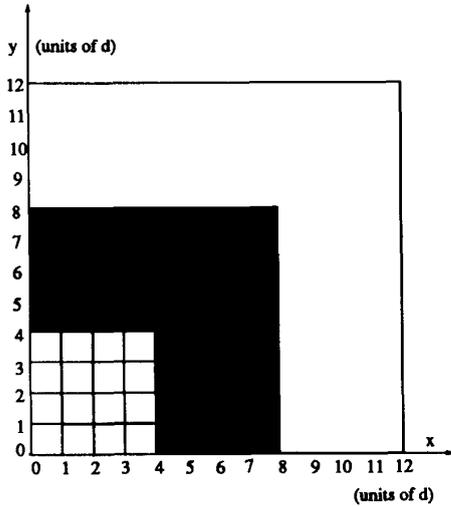


Fig. 5. Regular mask for a corner.

the transition regions:

$$f(a_{ij}; i, j = 1, \dots, N) = \sum_{k=1}^{N_s} (I(\vec{x}_k^+) - I(\vec{x}_k^-)) \quad (14)$$

where \vec{x}_k^+ is the k th sampling point along the contour D , \vec{x}_k^- is the k th sampling point along the contour C , N is the number of pixels in each dimension and N_s is the number of sampling points on contour C or D . Our problem can then be stated as: find a configuration $\{a_{ij}\}$ that maximizes f in (14) subject to the inequality constraints (10)-(13). We use the branch and bound algorithm [2] to solve this linear binary problem. Since we have described this algorithm in [3], we will not repeat it here.

We will present two examples of binary mask design under incoherent illumination using the branch and bound algorithm. The first one is a corner, shown in Fig. 5, and the other is an elbow, shown in Fig. 9. In both cases, the pixel size is chosen to be $0.2 \mu\text{m} \times 0.2 \mu\text{m}$, the feature size is $0.8 \mu\text{m}$. The bandwidth of the optical system is in units of cycle/ μm . For different values of bandwidth of the optical system, we obtain different optimized mask patterns, and for large values of bandwidth, the optimized patterns are identical to regular masks. This is consistent with the results we obtained in [3]. Let us define the parameter, relative transition width, as the ratio of the distance between the contours D and C to the feature size. We have found that for a given pixel size, thresholds, and bandwidth of the system, there is a minimum relative transition width for which we can find a solution by properly choosing the dose of the input pixel variables. As we would expect, for smaller bandwidth, or larger difference in the thresholds, the minimum transition width becomes larger.

Both examples shown in Figs. 7 and 11 correspond to optimal mask designs with minimum relative transition

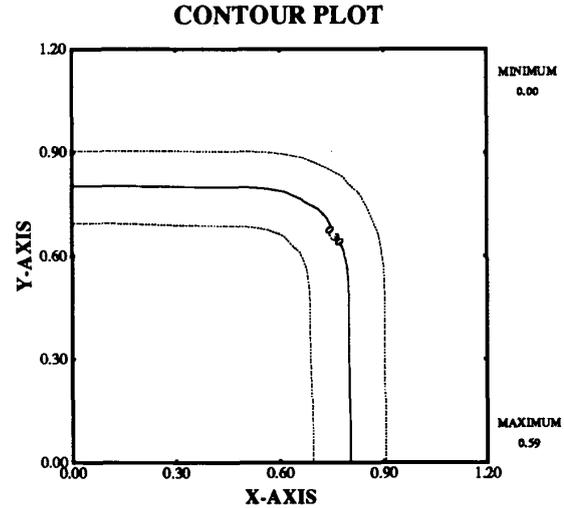


Fig. 6. Output intensity distribution due to the regular mask of Fig. 5 under incoherent illumination.

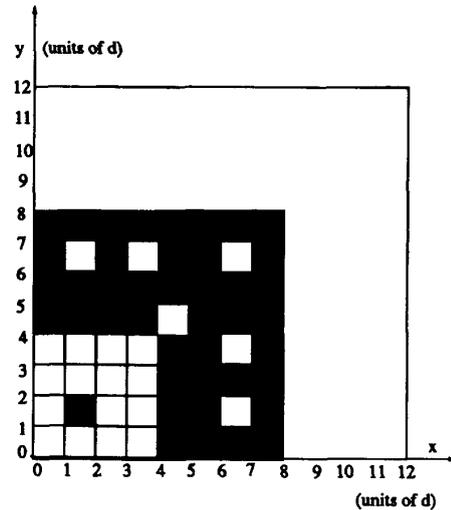


Fig. 7. Designed mask for a corner using the branch and bound algorithm.

width. Figs. 5 and 9 are desired patterns, or the regular mask patterns, Figs. 6 and 10 are image plots due to regular mask patterns, and Figs. 8 and 12 are image plots due to the optimal mask patterns of Figs. 7 and 11, respectively. We can see that in both cases, the output images due to optimal mask patterns have sharper 0.3 threshold contour shape than those of regular masks. We believe the low contrast at the corners in Figs. 8 and 12 to be inherent to incoherent illumination, for which there is a trade off between contrast and sharpness of contour shape. After all, these plots are from a system which allows minimum amount of frequency components to pass through. With the introduction of more frequency com-

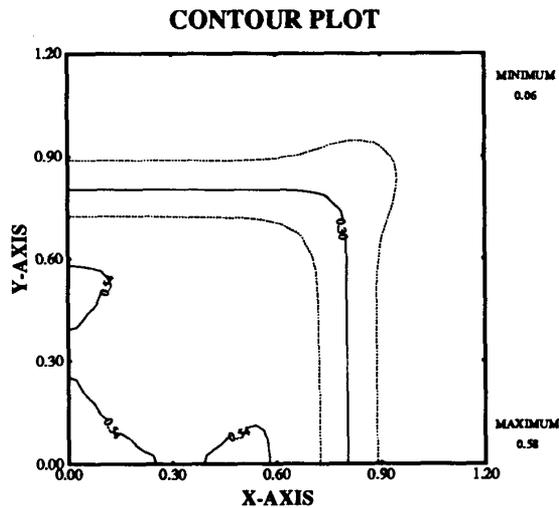


Fig. 8. Output intensity distribution due to the designed mask of Fig. 7 under incoherent illumination.

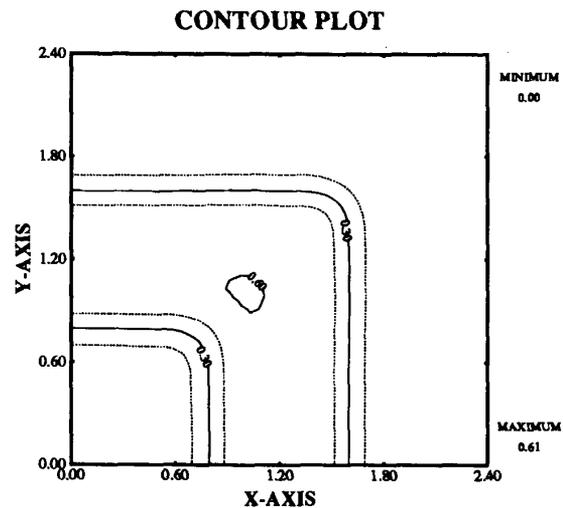


Fig. 10. Output intensity distribution due to the regular mask of Fig. 9 under incoherent illumination.

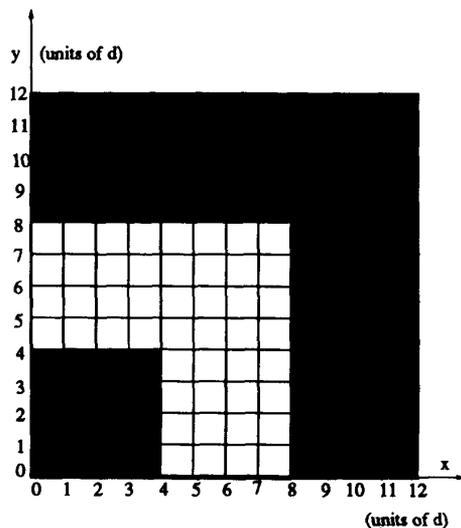


Fig. 9. Regular mask for an elbow.

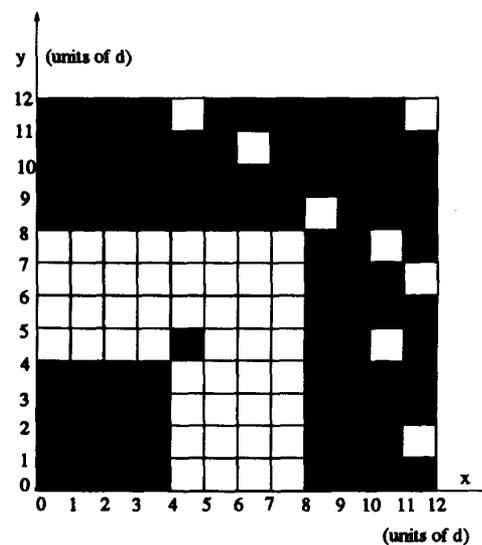


Fig. 11. Designed mask for an elbow using the branch and bound algorithm.

ponents or some coherence, the contrast can be increased, as we will show in later examples. Note that, unlike the desired pattern, the solution in Fig. 12 is not symmetric with respect to the $x = y$ line. We will discuss this later in Section VI.

Compared to the Mean Squared Error (MSE) criteria formulation [3], the above specification has several advantages. First, it results in better intensity contours, hence it is more appropriate for optical lithography application. Second, it results in a simpler formulation than MSE, allowing us to solve larger problems. Thirdly, these specifications have incorporated a simple model of photoresist development, which allows us to take advantage

of the contrast-enhancement characteristics of photoresist.

The branch and bound algorithm can also be applied to imaging with partially coherent illumination, and to phase-shifting mask design. In fact, the partially coherent illumination case requires quadratic rather than linear programming algorithm and the phase-shifting case involves a ternary programming problem with pixel values $\{-1, 0, 1\}$ rather than a binary programming one with values $\{0, 1\}$. Nonetheless, the implementation is more involved than that of the incoherent case, and therefore we have opted to use the simulated annealing algorithm instead.

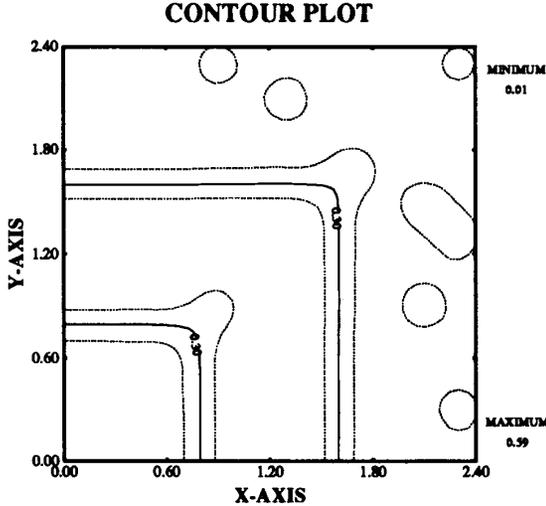


Fig. 12. Output intensity distribution due to the designed mask of Fig. 11 under incoherent illumination.

V. OPTIMIZATION BY SIMULATED ANNEALING FOR PHASE-SHIFTING MASKS

A. Formulation

In this section, we explore the use of the Simulated Annealing (SA) algorithm for phase shifting mask design with partially coherent illumination. To use this algorithm, we have converted all the constraints in inequalities (10)–(13) into a total objective function. The total objective function is chosen to be a weighted sum of local objective functions corresponding to different sampling points and is given by

$$H(\vec{a}) = \sum_{k \in N_A} h_A(I(\vec{x}_k, \vec{a})) + \sum_{k \in N_B} h_B(I(\vec{x}_k, \vec{a})) + \sum_{k \in N_C} h_C(I(\vec{x}_k, \vec{a})) + \sum_{k \in N_D} h_D(I(\vec{x}_k, \vec{a})) \quad (15)$$

where k is the index for sampling points, \vec{a} is the vector of pixel variables, N_i is the set of indices of the samples in region i with $i = A, B, C$ or D . This definition of index i will be used to define other related variables. The objective function $h_i(I(\vec{x}_k, \vec{a}))$, at a particular sampling point with coordinate \vec{x}_k , is chosen to be a nonlinear function of the error at that point,

$$h_i(I(\vec{x}_k, \vec{a})) = \begin{cases} e_i(I(\vec{x}_k, \vec{a})) & \text{if } e_i(I(\vec{x}_k, \vec{a})) < 0 \\ \lambda_i e_i(I(\vec{x}_k, \vec{a})) & \text{if } e_i(I(\vec{x}_k, \vec{a})) \geq 0 \end{cases} \quad (16)$$

where λ_i is the penalty factor which reflects the relative importance of each constraint in inequalities (10)–(13) and the error function $e_i(I(\vec{x}_k, \vec{a}))$ at each sampling point is

defined as follows:

$$e_i(I(\vec{x}_k, \vec{a})) = \begin{cases} I(\vec{x}_k, \vec{a}) - a & \text{if } \vec{x}_k \in A \\ b - I(\vec{x}_k, \vec{a}) & \text{if } \vec{x}_k \in B \\ I(\vec{x}_k, \vec{a}) - c & \text{if } \vec{x}_k \in C \\ d - I(\vec{x}_k, \vec{a}) & \text{if } \vec{x}_k \in D \end{cases} \quad (17)$$

The error function $e_i(I(\vec{x}_k, \vec{a}))$ in (17) is chosen in accordance with the inequalities (10)–(13). Specifically since $I(\vec{x})$ is specified to be smaller than a and c in inequalities (10)–(13) respectively, the form of $e_i(I(\vec{x}_k, \vec{a}))$ for $\vec{x}_k \in A$ and $\vec{x}_k \in C$ in (17) is chosen in such a way as to force e_i to be as negative as possible. A similar argument can be made for $\vec{x}_k \in B$ and $\vec{x}_k \in D$. Therefore the more positive (negative) the error function is, the more (less) it violates the constraints in inequalities (10)–(13). Since the SA algorithm does not depend on the functional form of the objective function, we have chosen the objective function (16) to be a nonlinear function of the error function. In fact, this choice of the form of objective function and the penalty factors described later enables us to apply SA algorithm to mask design problems.

The penalty factors are chosen such that whenever the constraints in inequalities (10)–(13) are violated at any sampling point by a certain small amount, the increase in the objective function due to that point will be larger than the sum of all the negative contributions from those sampling points at which the constraints in inequalities (10)–(13) are satisfied, i.e., $e_i < 0$. Therefore the objective function is dominated by the constraint violating points. This way, the elimination of constraint violating points becomes the highest priority of the optimization process. Only when all the constraints are satisfied, does the optimization scheme start paying attention to the improvement of constraint satisfaction. To achieve the above goal, we have chosen the penalty factors as follows:

$$\lambda_i = \lambda_i' K \quad (18)$$

where $\lambda_i' > 1$ is the penalty factor which reflects the relative importance of each region or contour, and K is a large positive constant.

B. Application of Simulated Annealing

Simulated Annealing is a technique developed in recent years to solve complex optimization problems [7]–[12]. It is based on the analogy between the simulation of the annealing of solids and the problem of solving large combinatorial optimization problems. Here we provide a brief description of the basic algorithm and concentrate on the way we apply it to the mask design problem. The reader is referred to [7], [13], [10], [11] for a complete description of the SA algorithm. For convenience, we assume binary optimization in the following, unless otherwise indicated.

In combinatorial optimization problems, if we assume configurations to correspond to states of a solid, the objective function H to energy, and the control parameter T to temperature in physics, then the annealing process can be simulated on a computer by assigning the Boltzmann distribution to the configurations according to the objective function. The annealing process in physics for finding the ground state of a solid is then equivalent to finding the set of configurations which result in minimum value of objective function in combinatorial optimization.

Consider different binary patterns of the mask, denoted by a random vector X , to be the state of a statistical system. If the objective function, or energy function corresponding to a state X is $H(X)$, then according to Boltzmann distribution the probability of being in state X at thermal equilibrium is

$$P\{X = X\} = \frac{1}{Z(T)} e^{-[H(X)/T]}, \quad (19)$$

where Z is a normalizing constant, or the partition function in statistical physics. At high temperature, this distribution is almost uniform, and the system is equally likely to be in any state. We gradually decrease the temperature parameter T , allowing the system to reach thermal equilibrium at each T . As the temperature decreases, the Boltzmann distribution concentrates more on the states with low energy. Finally, when the temperature approaches zero, all other states except for the minimum energy states have a zero probability of occurrence, and a ground state is found. However, if the temperature is decreased too fast, the system may get trapped in some local minimum of energy, as in the case of annealing solids, where defects occur. In computer simulations, we carefully choose a sequence of decreasing temperatures, $\{T_m\}$, with $T_m \rightarrow 0$. Starting from an initial mask pattern, at each T_m a sequence of mask patterns or states is generated by randomly choosing a pixel and flipping it to the other state. This is called a proposal for a transition. If we denote the state before the flip to be state i , the state after the flip to be state j , and the resulting change of energy by $\Delta H_{ij} = H(X_j) - H(X_i)$, then the probability for state j to be the next state in the sequence is chosen to be 1, if $\Delta H_{ij} < 0$, and $\exp(-(\Delta H_{ij}/T_m))$, if $\Delta H_{ij} \geq 0$. Thus there is a nonzero probability of continuing with a mask pattern with higher energy than the current state. This process is continued until equilibrium is reached, i.e., until the probability distribution of the system approaches the Boltzmann distribution, given by (19). We notice that the probability of making a transition from state i to state j does not depend on the history of how state i was reached, hence the generated sequence $\{X_m(n)\}$ for each T_m is a Markov chain, [14].

The temperature is then lowered in steps, with the system being allowed to approach equilibrium for each step of T_m by generating a sequence of mask patterns in the

previously described way. The algorithm is terminated for a small value of T_m , for which virtually no deterioration is accepted anymore. The final 'frozen' state, or mask pattern, is then taken as the solution to the mask optimization problem.

For phase-shifting masks, the values each pixel variable can take are $\{-1, 0, 1\}$. We still randomly choose a pixel; however, there are two proposed transitions instead of one for each pixel. An acceptance rule that decides which of the three states may be the next state in the sequence has to be chosen. The proposed transition is considered rejected if the pixel value is not changed according to the acceptance rule. The proposed transition is considered accepted if proposed pixel variable takes one of the other two values different from its original value, according to the acceptance rule.

The acceptance rule can be chosen to be the conditional probability of state j , given the previous state i and pixel k being chosen to be perturbed. According to Boltzmann distribution, this conditional probability is

$$A_{ij}(T) = \frac{\exp(-H(\vec{a}_j)/T)}{\sum_{l \in \omega} \exp(-H(\vec{a}_l)/T)} \quad (20)$$

$$= \frac{\exp(-\Delta H_{ij}/T)}{\sum_{l \in \omega} \exp(-\Delta H_{il}/T)} \quad (21)$$

where l is the index for possible next states, ω is the set of three states generated by keeping all other pixel values the same as in state i except for pixel k , which takes on values $\{-1, 0, 1\}$, and ΔH_{il} is the change of energy function from state i to state l .

Since the light intensity due to a pixel dies out with distance, and the interaction between two pixel light sources decreases with their distance, the light intensity and hence the energy function at any sampling point is a function of only its neighboring pixels, say in a circular region centered at the sampling point with radius R . For the same reason, any pixel can influence the intensity only at sampling points in a circular region centered at the pixel with radius R . The acceptance probability of a change of value of pixel k depends on the change of energy function ΔH_{il} , which only depends on the light intensity at sampling points in a neighborhood of pixel k . The intensity at those sampling points in turn only depends on values of pixels in their neighborhood. Therefore, the acceptance probability only depends on the values of pixels in a region centered at pixel k with radius $2R$. This can be used to save computation time.

The choice of the initial temperature, rule for decreasing temperature, length of Markov chains, and a stopping criterion for the entire SA algorithm is called a *cooling schedule*. The cooling schedule has a great deal of influence on the performance of the SA algorithm. Before describing our cooling schedule, we need to introduce the

concept of *quasi-equilibrium*: let L_m be the length of the m th Markov chain, $\vec{b}(L_m, T_m)$ and $\vec{q}(T_m)$ be the vectors representing the probability distribution after L_m transitions and the stationary distribution of the homogeneous Markov chain at the value T_m respectively. An annealing algorithm is said to be in quasi-equilibrium at the value of control parameter T_m , if $\|\vec{b}(L_m, T_m) - \vec{q}(T_m)\| < \epsilon$ for the required precision ϵ . The definition of the metric $\|\cdot\|$ affects the cooling schedules. The following discussion describes the particular cooling schedule that we have chosen to use:

Initial Value of Control Parameter: The initial value of T is chosen such that all transitions are equally likely to be accepted, ignoring the differences in cost function. As a quantitative measure, we define the *acceptance ratio* χ as the number of accepted transitions divided by the number of proposed transitions. We choose the initial value χ_0 and determine T_0 by calculating the average increase in cost function over all proposed transitions, ΔH^+ , for a number of random transitions and set T_0 to be $T_0 = (\Delta H^+ / \ln [1/\chi_0])$, [15].

Final Value of the Control Parameter: This is the stopping criterion for the entire SA algorithm. A reasonable choice would be to terminate the execution of the algorithm if the last states of consecutive Markov chains are identical for a number of chains [16].

Length of Markov Chains: An intuitive choice which is based on the concept of quasi-equilibrium was made by Skiscim and Golden [17]: define an *epoch* as a number of transitions with a fixed number of acceptances and the energy of an epoch as the energy value of the last state of an epoch. As soon as the energy of an epoch is within a specified distance from the energy of one of the preceding epochs, the Markov chain is terminated. This way, the termination of a Markov chain is related to the fluctuations of the values of the energy function in that chain.

Decreasing the Control Parameter: There is a trade-off between the speed of reducing the control parameter T_m and the length L_m of the Markov chain at that particular T_m . The convergence of SA algorithm is based on consecutive transitions from one equilibrium to another equilibrium. In computer simulations, the concept of quasi-equilibrium has to be used. A Markov chain is terminated when quasi-equilibrium is achieved at the value of control parameter T_m . When T_m is decreased to T_{m+1} , the quasi-equilibrium is disturbed. A new sequence of Markov chain is constructed to achieve a new quasi-equilibrium. For larger decreases in T_m , the previous quasi-equilibrium is disturbed more seriously, and the length L_{m+1} of the Markov chain has to be longer in order to re-establish quasi-equilibrium. For this reason, one usually opts to decrement T_m slowly. A number of authors have used geometric sequence for T_m ,

$$T_m = \alpha T_{m-1}, \quad m = 1, 2, \dots \quad (22)$$

where α is a constant smaller than but close to 1. Choices

of α have ranged from 0.5 to 0.99 by different authors [15], [11]. We have chosen $\alpha = 0.9$ in our simulation.

For more details on cooling schedule, refer to [7]–[13], [15], [17].

It should be noted that the iterative algorithm, in which a transition is accepted (rejected) whenever it results in a reduction (increase) of objective function, is simply a *greedy* annealing algorithm in which the temperature is reduced to zero at the very beginning of the annealing process. While the iterative algorithm is likely to converge to local rather than global minima, it is less computation intensive than the SA algorithm. Therefore it could potentially result in acceptable mask designs if the initial patterns are chosen close to the global minimum.

C. Simulation Results

In the following simulations, (1) is used to calculate output intensity and SPLAT [6] is used to calculate the TCC function and to check the final results. The TCC function is calculated only once for each optimization. The change of intensity due to one pixel change is calculated only in a neighborhood of that pixel in order to speed up computation. After every 20 accepted transitions, the entire intensity profile is re-calculated to prevent accumulation of errors due to the approximation in each transition. In all the examples in this section, four fold symmetrical patterns are considered. The patterns shown in the figures are only the first quadrant of the actual patterns. The common parameters for the examples are as follows: the wavelength is $0.4358 \mu\text{m}$, the numerical aperture is 0.28, All the space lengths are in unit of μm , the pixel size is 0.2×0.2 and the feature size is 0.8. We show designs from iterative method along with simulated annealing in the following examples.

Design examples of a corner and an elbow are shown in Figs. 13 through 28. The white regions represent transparent area, light shaded regions represent 180 degree phase shift area without damping, and the dark regions represent opaque area. Figs. 13 and 15 show the desired binary image for a corner and an elbow respectively. These figures also represent the ‘‘regular’’ masks that would have been used in situations where no pre-distortion is included. Their corresponding output images are shown in Figs. 14 and 16.

Examples of binary mask design for the corner of Fig. 13 are shown in Figs. 17 and 19. The design parameters used are as follows: the relative transition width is 0.02, the thresholds are $a, b, c, d = 0.15, 0.70, 0.20, 0.30$ respectively, and the penalty factors are chosen to be $\lambda'_{A,B,C,D} = 5, 5, 10, 10$ respectively. As we can see, the choice of penalty factors $\lambda'_{A,B,C,D}$ has emphasized contour shape more than contrast; therefore the contours due to designed masks are sharper than those due to regular masks in Fig. 14. However, the contrast in Fig. 18 is seriously compromised, whereas the contrast in Fig. 20 is

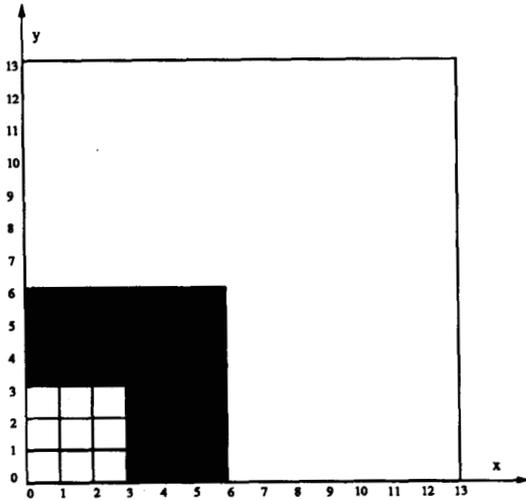


Fig. 13. Regular mask for a corner.

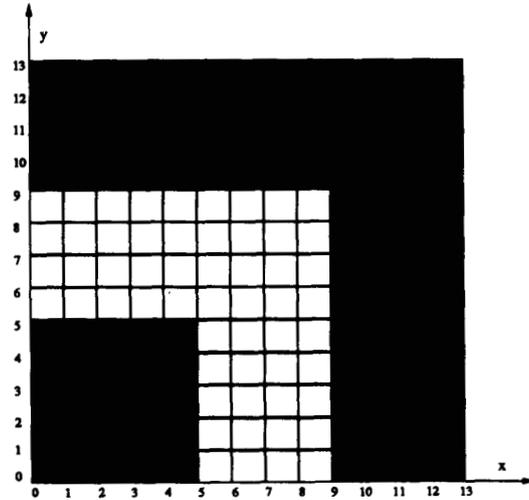


Fig. 15. Regular mask for an elbow.

CONTOUR PLOT

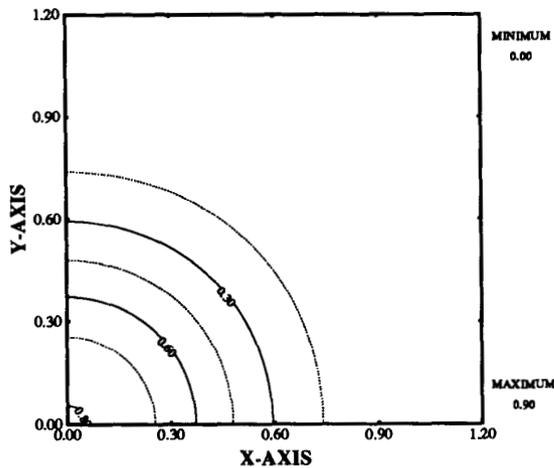


Fig. 14. Output intensity distribution due to regular mask of Fig. 13 under partially coherent illumination.

CONTOUR PLOT

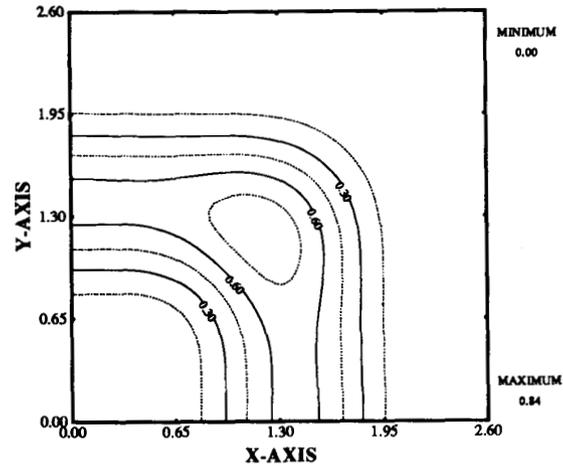


Fig. 16. Output intensity distribution due to regular mask of Fig. 15 under partially coherent illumination.

still comparable to Fig. 14. These observations indicate that our design techniques are effective and that simulated annealing is more effective than iterative method, as we would expect.

Examples of phase-shifting mask design are shown in Figs. 21 through 27. The design examples for the corner of Fig. 13 are shown in Figs. 21 and 23, where the relative transition width is 0.1, the thresholds are $a, b, c, d = 0.15, 0.70, 0.20, 0.30$, and the penalty factors are chosen to be $\lambda'_{A,B,C,D} = 10, 10, 5, 5$, respectively. Due to the loose requirement on the transition width and the emphasis on domain constraints, the intensity profiles in Figs. 22 and 24 have better contrast than those in Figs. 18 and

20, but not as good contour precision, or linewidth control. To achieve a proper balance, the penalty factors need to be accordingly modified. The design from SA algorithm again outperforms that of the iterative method. Design examples for the elbow of Fig. 15 are shown in Figs. 25 and 27, where the relative transition width is 0.02, the thresholds are $a, b, c, d = 0.15, 0.70, 0.20, 0.30$, and the penalty factors are chosen to be $\lambda'_{A,B,C,D} = 5, 5, 5, 5$ respectively. Again, the output image due to the mask from SA algorithm compares favorably to that of iterative method which in turn compares favorably to that of the regular mask in terms of both contrast and contour shape.

It is interesting to note that there are many auxiliary

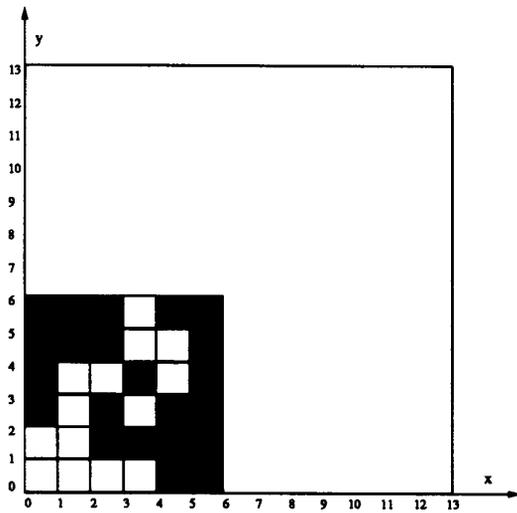


Fig. 17. Designed binary mask for the corner of Fig. 13 by iterative method.

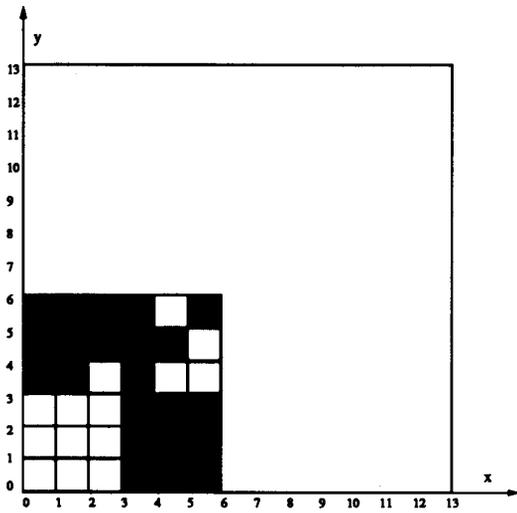


Fig. 19. Designed binary mask for the corner of Fig. 13 by simulated annealing.

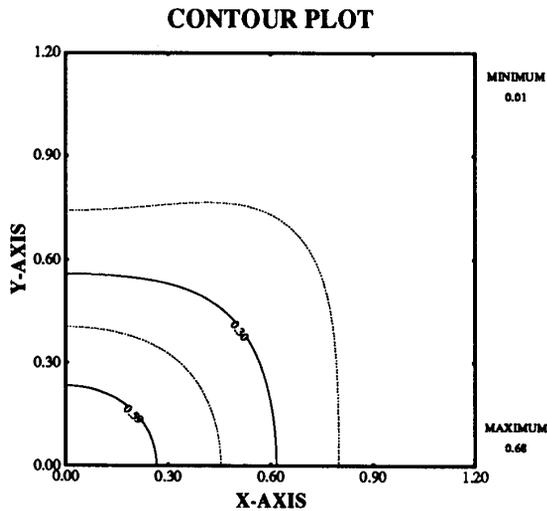


Fig. 18. Output intensity distribution due to the designed binary mask of Fig. 17 under partially coherent illumination.

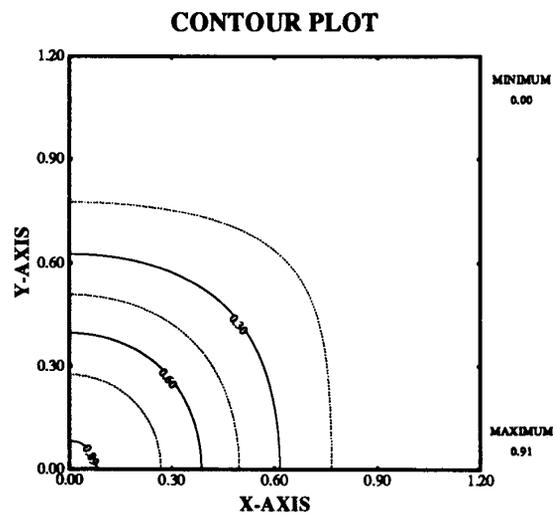


Fig. 20. Output intensity distribution due to the designed binary mask of Fig. 19 under partially coherent illumination.

features in region *A* which are not printed. A feature is referred to as not printed if the light intensity at its location is below threshold α . Although the designed masks look complex, some general observations can be made. Generally, in order to increase the light intensity in region *B* without changing the exposure time for large features, the auxiliary non-phase-shifting pixels are placed around and nearby the feature. Then some phase-shifting pixels are placed nearby those auxiliary pixels to reduce the intensity in order to force the transition to be in between contours *C* and *D*, or to prevent auxiliary pixels from printing. Sometimes, even consecutive phase-shifting

pixels are needed to cancel the intensity extending out of the *B* region. Our approach has taken full advantage of the contrast enhancement characteristic of the photoresist, i.e., as long as the local intensity is less than 0.15, the image is not printed on the photoresist. The auxiliary pixels far from transition regions may impose some difficulty in practice. However, we can eliminate those pixels by choosing their values to be fixed in the optimization. This can also speed up the algorithm, though, the designed masks would of course be less optimal.

There are two other important aspects the reader should be aware of. One is that since the numerical aperture used

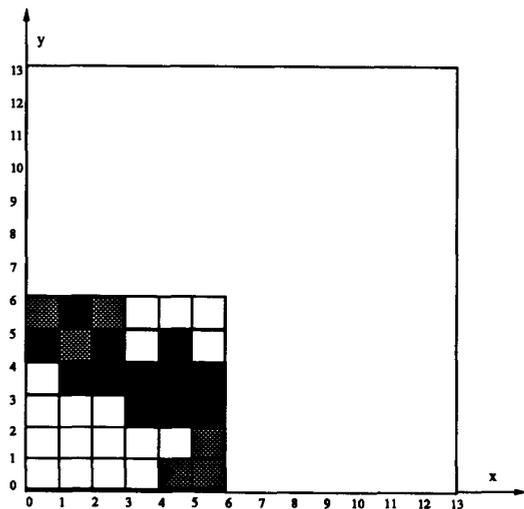


Fig. 21. Designed phase-shifting mask for the corner of Figure 13 by iterative method.

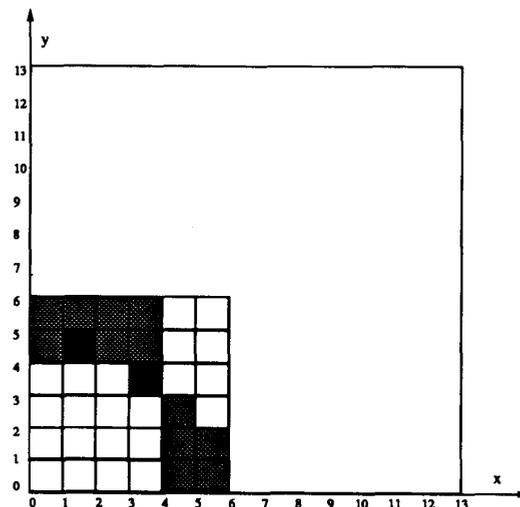


Fig. 23. Designed phase-shifting mask for the corner of Fig. 13 by simulated annealing.

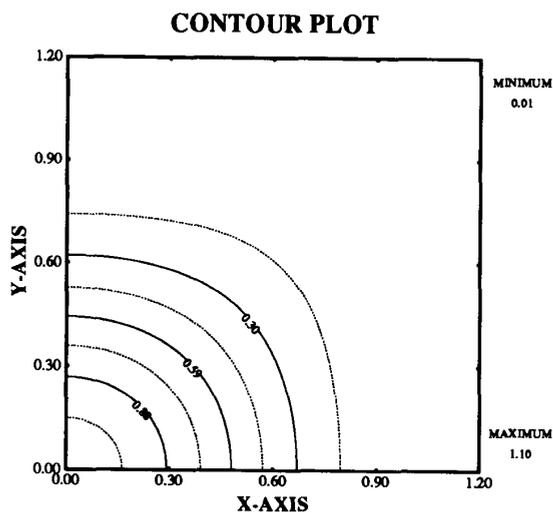


Fig. 22. Output intensity distribution due to the designed phase-shifting mask of Fig. 21 under partially coherent illumination.

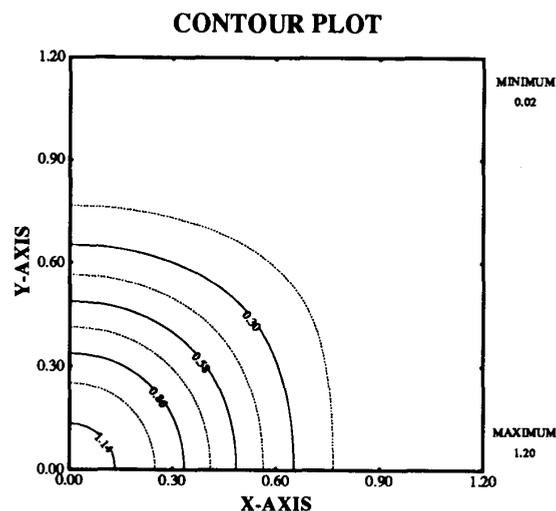


Fig. 24. Output intensity distribution due to the designed phase-shifting mask of Fig. 23 under partially coherent illumination.

here is quite small, i.e., 0.28, depth of focus is not an issue here. This can be verified by simply computing the aerial image at maximum defocus plane. The other is that the degree of coherence used here, $s = 0.7$, results in relatively low interference. When s is reduced, the improvement of optimally designed mask over conventional masks will be considerably larger.

D. Masks with Phase Conflicts and Portability

Among all the phase-shifting masks proposed so far, alternating phase-shift design, first proposed by Levenson [18], enjoys the most improvements over conventional

masks. Unfortunately, it results in phase conflicts when applied to features other than periodic line-space patterns. Our proposed technique in this paper provides a unique approach to solving this problem, which is generally considered to be a bottle-neck in phase-shifting mask application.

Figs. 29 through 36 show an example of phase-conflicting mask design. The pictures are periodic in both x and y directions. The simulation is done for i -line ($0.365 \mu\text{m}$) stepper with 0.32 NA on Waycoat HiPR 6512 resist. Again, all the lengths are in units of μm . For masks in Figs. 29 and 32, the dark region represents opaque, white region represents transparent, dark shaded region repre-

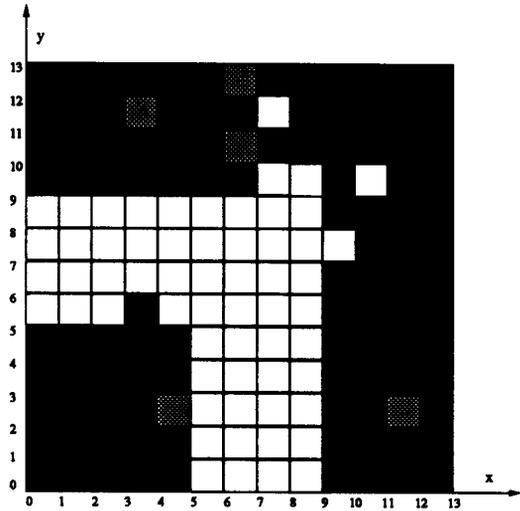


Fig. 25. Designed phase-shifting mask for the elbow of Fig. 15 by iterative method.

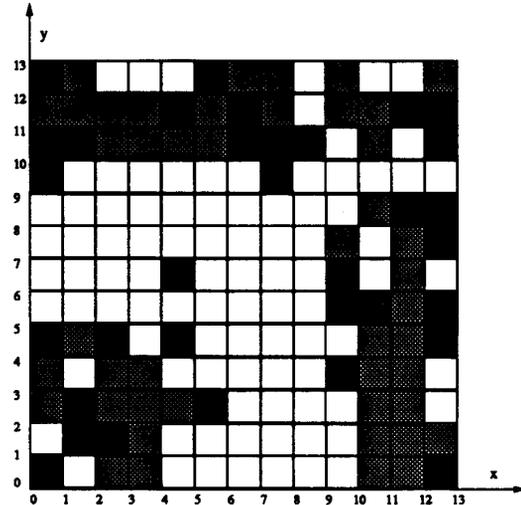


Fig. 27. Designed phase-shifting mask for the elbow of Fig. 15 by simulated annealing.

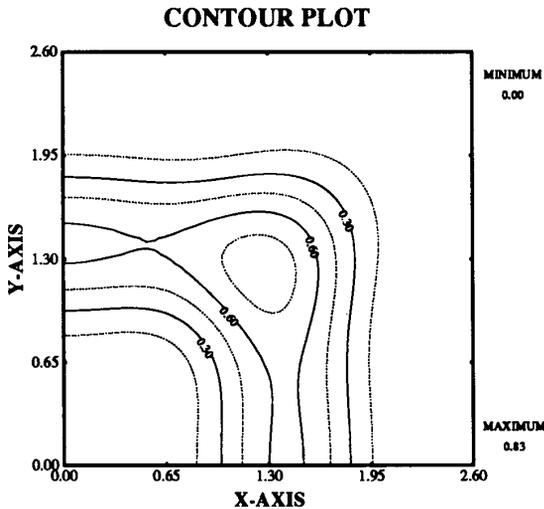


Fig. 26. Output intensity distribution due to the designed phase-shifting mask for the elbow by iterative method.

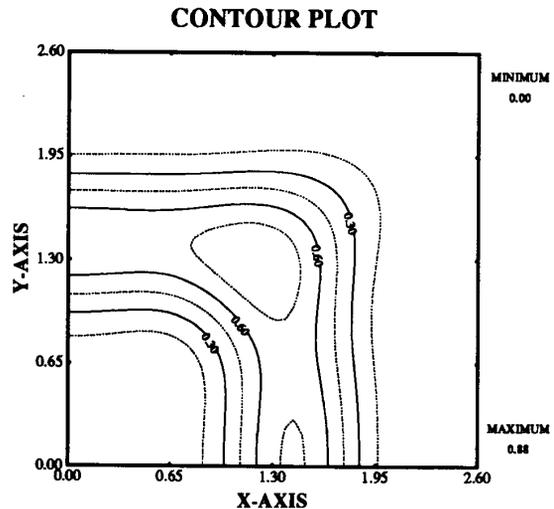


Fig. 28. Output intensity distribution due to the designed phase-shifting mask for the elbow by simulated annealing.

sents 180° phase-shift transparent and light shaded region represents 90° phase-shift transparent. The images in Figs. 31 and 34 are projection of photoresist profiles after development simulation by SAMPLE3D, a 3-D resist development simulation program developed at University of California, Berkeley. In this example, the goal is to print a cross on the resist. Suppose phase assignment at nearby features lead to a phase-conflict which requires us to connect a 0° and a 180° line. An intuitive solution [19], [20] is to place a 90° phase line in between the 0° and 180° regions, as shown in Fig. 29. The resulting intensity distribution of such an approach is shown in Fig. 30. As seen in Fig. 31, this results in an unsuccessful connection. Our

optimally designed mask, its resulting intensity distribution and developed resist profile are shown in Figs. 32, 33, and 34 respectively. As seen in Fig. 34, the optimal mask results in full connection without "bridging" different features. Figs. 35 and 36 are the corresponding 3-D developed resist profiles of Figs. 31 and 34, respectively. This example also serves to verify that the method of comparing the aerial images in our previous examples is valid.

In the above example, we have implicitly introduced a new concept which we refer to as "optimization under an environment." That is, only the center portion of the mask is optimized, and the surrounding portions are left intact.

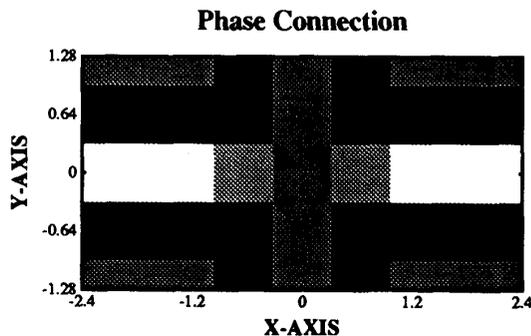


Fig. 29. A simple $0^\circ/180^\circ$ phase connection with 90° phase. Feature size and connection length are both $0.64 \mu\text{m}$.

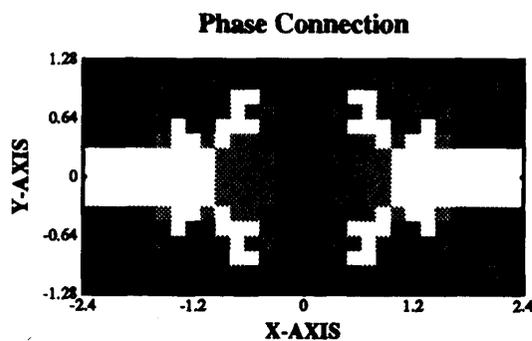


Fig. 32. Designed 3-level phase-shifting mask for the phase connection. The pixel sizes are $0.16 \times 0.16 (\mu\text{m})^2$.

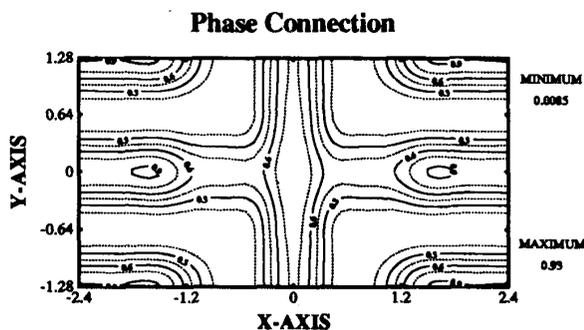


Fig. 30. Output intensity distribution due to the simple mask of Fig. 29.

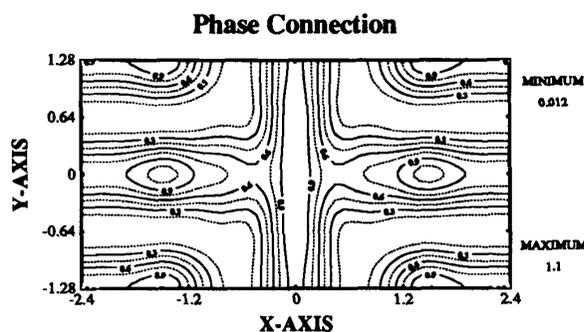


Fig. 33. Output intensity distribution due to the designed mask of Fig. 32.

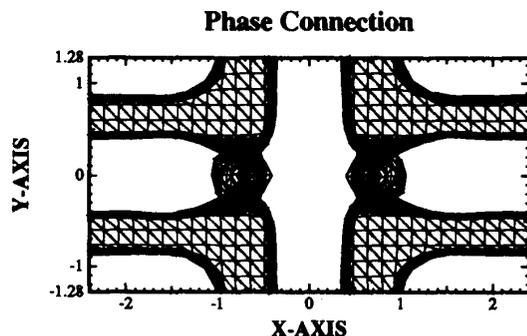


Fig. 31. Projection of the resist profile on the chip after photoresist development for the simple mask of Fig. 29.

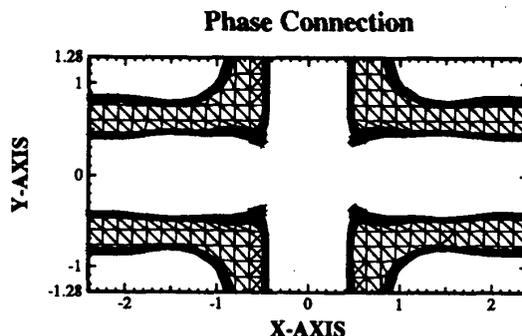


Fig. 34. Projection of the resist profile on the chip after photoresist development for the designed mask of Fig. 32.

This way, the interference between light through the optimized portion of the mask and nearby unoptimized portion of the mask are already taken into account in our optimization. Therefore the optimized mask, together with the environment under which it is optimized, become a portable design unit. The unit can be applied whenever the environment matches. This arrangement allows us to build an optimized phase-shifting mask library. The design of phase-shifting mask then becomes a simple task consisting of two steps. First, assign the phase of features alternatively without paying attention to the resulting phase-conflicts; then choose the appropriate optimized

units from an optimized mask library to replace the phase-conflicting regions resulting from the first step.

VI. DISCUSSION

It is worthwhile to note that except for the mask shown in Fig. 7, all the remaining designed masks in this paper are unsymmetric about the line $x = y$. These designs result in visible unsymmetric intensity patterns except in Figs. 20 and 24. This can be partially explained by noting that symmetry enforcement merely reduces the size of the solution space for the SA algorithm. Another explanation

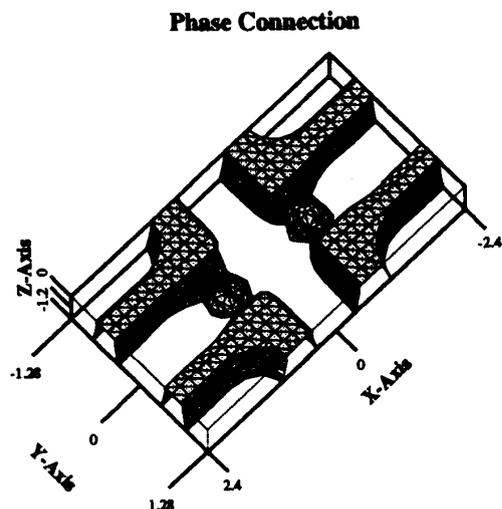


Fig. 35. Developed resist profile from the simple mask in Fig. 29.

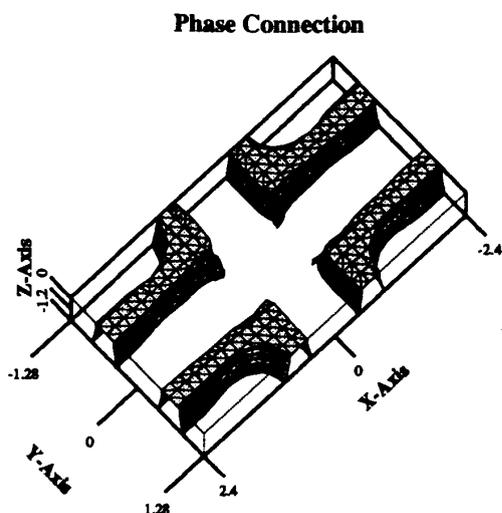


Fig. 36. Developed resist profile from the designed mask in Fig. 32.

is that enforcing symmetry requires simultaneous flipping of two, rather than one, pixels at a time, and therefore results in twice as much change in the light intensity. Thus, unsymmetric masks alter the intensity profile in finer steps than symmetric ones and hence are more likely to satisfy the design specifications. Another observation to be made is that an unsymmetric solution and its mirror image with respect to the diagonal, result in the same value of the objective function. Thus the symmetry of the problem manifests itself through degeneracy.

The iterative algorithm takes about 0.5 to 2 hours for the elbow design on a SUN SPARC-1 workstation. The simulated annealing algorithm takes from 6 to 9 hours to optimize the elbow masks. Although we can use more sophisticated cooling schedules to speed up the algorithm,

we have found that computing the output intensity is the most computationally intensive part of the optimization, and that the convergence of the algorithm is not a major issue. So, if we find a formula for a single pixel perturbation or if we find a fast way to compute the functions $G_{ijlm}(x, y)$, then optimization speed can be drastically increased. The generation probability could also be made nonuniform in order to speed up the convergence of the algorithm, exploiting the fact that the edge pixels are more important than other pixels.

Comparing the branch and bound algorithm and simulated annealing algorithm, we notice that the branch and bound algorithm is more sophisticated in that it can find absolute minimum or maximum, and the threshold constraints are completely satisfied for any obtained solution; but its implementation is more involved than SA algorithm. In particular, its implementation for the case of partially coherent illumination depends on the functions $G_{ijlm}(x, y)$, that is, it depends on explicit functional form. While simulated annealing algorithm is easy to implement, does not depend on the form of the objective function, and the relation between pixel variables and the objective function can be an implicit function, it does not guarantee that all the constraints are satisfied; furthermore, we need to choose proper weights on sampling points in order to obtain acceptable results, the computation time could be long and the cooling schedule needs to be fine tuned. Nonetheless, SA formulation is very versatile, and can handle many more factors than what we have dealt with here. Besides, the iterative method can be used whenever a quick solution, rather than near global minimum is desired. Comparing the output intensity due to the iterative algorithm and simulated annealing algorithm in Figs. 18 and 20, Figs. 22 and 24, and Figs. 26 and 28, we notice that the later outperform the former either in terms of contour sharpness or contrast. This indicates the necessity of avoiding the local minima in the optimization process.

We notice that the designed masks are very different from conventional masks. The introduction of phase-shifting mask provides more degrees of freedom, but the determination of predistorted masks has also become more complex. For the phase-conflicting case we solved here, it is unimaginable for a person, however experienced he/she is, to arrive at a result by trial and error process. To make full use of phase-shifting masks, experience and existing approaches are not enough, and new systematic techniques such as the ones proposed in this paper are needed. We believe that a combination of Levenson type and locally optimized phase-shifting mask as proposed here is a viable candidate for future phase-shifting mask design.

The bit-map representation of our designed masks may present difficulties in mask making due to the large amount of data to be processed. Besides, misalignment may affect our designs more than simpler designs. Therefore, our proposed algorithm is most useful for small regions, es-

pecially for situations where simple designs are inadequate. Examples of such cases are bias for small features, or connecting different phases. In addition, our algorithm can be used as a tool to improve our understanding of phase shifting mask design. Specifically, the intuition gained from optimum designs can be used for simpler, practical designs. From a theoretical viewpoint, our algorithm provides an estimate of the upper bound of the performance of pre-distorted masks. Finally, our algorithm can be used to design masks that compensate distortion due to optical aberrations, and can be extended to E-beam and X-ray projection printing.

While we have verified our designs via simulation packages such as SAMPLE3D, it needs to be further verified experimentally. Possible sources of discrepancy between our simulation and experimental results might arise due to the fact that our algorithm is based on the intensity threshold criterion, and does not consider the distortion of light at defocus. Specifically, it optimizes intensity at the focus plane, and SAMPLE3D propagates the intensity profile at focus plane down into photoresist assuming plane wave and normal incidence. Therefore, our proposed design technique is likely to perform better for top surface imaging technologies than single layer processes. Our preliminary results on incorporating defocus effects in the optimization process are encouraging [21].

ACKNOWLEDGMENT

The authors are grateful to Professor Andrew Neureuther for his generous help in terms of intuitive discussions and advice on lithography aspects of this project. The authors also thank Professor William Oldham for his comments on practical aspects of e-beam mask making and Kenny Toh for providing the lithography imaging program SPLAT and photoresist development program SAMPLE3D.

REFERENCES

- [1] E. A. B. Saleh, "Image synthesis: Discovery instead of recovery," in *Image Recovery-Theory and Application*, ch. 12, Henry Stark, Ed. Orlando, FL: Academic Press, 1987.
- [2] A. H. Land and S. Powell, *Fortran Codes for Mathematical Programming: Linear, Quadratic and Discrete*. New York: Wiley, 1973.
- [3] Y. Liu and A. Zakhor, "Optimal binary image design for optical lithography," in *SPIE Optical/Laser Microlithography III*, vol. 1264, pp. 401-412, 1990.
- [4] H. H. Hopkins, "On the diffraction theory of optical images," in *Proc. Royal Soc. Series A.*, vol. 217, no. 1131, pp. 408-432, 1953.
- [5] M. Born and E. Wolf, *Principles of Optics*. London: Pergamon, 1980.
- [6] K. K. H. Toh, "Two-dimensional images with effects of lens aberrations in optical lithography," Memorandum UCB/ERL M88/30, University of California, Berkeley, May 20, 1988.
- [7] E. H. L. Aarts and P. J. M. van Laarhoven, "A pedestrian review of the theory and application of the simulated annealing algorithm," in *Heidelberg Colloquium on Glassy Dynamics and Optimization*. Berlin: Springer Verlag, Spinger Lecture Notes, 1987.
- [8] S. Gres, C. M. Soukoulis, and K. Levin, "Cooling-rate dependence for the spin-glass ground-state energy: Implications for optimization by simulated annealing," *Phys. Rev. Lett.*, vol. 56, pp. 1148-1151, 1986.
- [9] B. Hajek, "A tutorial survey of theory and applications of simulated annealing," in *Proc. 24th Conf. on Decision and Control*, Ft. Lauderdale, Dec. 1985, pp. 775-760.
- [10] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671-680, 1983.
- [11] P. J. M. van Laarhoven and E. H. L. Aarts, *Simulated Annealing: Theory and Applications*. Boston: D. Reidel Publishing, 1987.
- [12] D. Mitra, F. Romeo, and A. Sangiovanni-Vincentelli, "Convergence and finite-time behavior of simulated annealing," in *Proc. 24th Conf. on Decision and Control*, Ft. Lauderdale, FL, Dec. 1985, pp. 761-767.
- [13] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 721-741, 1984.
- [14] A. J. Thomasian, "Random process," Dept. of Electrical Engineering and Computer Science, University of California, Berkeley, 1990, text manuscript.
- [15] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, "Optimization by simulated annealing: An experimental evaluation," in *Workshop on Statistical Physics in Engineering and Biology*, Yorktown Heights, NY, April 1984, revised version, 1986.
- [16] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Tech. Rep. RC 9355*, IBM Research Rep., 1982.
- [17] C. C. Skiscim and B. L. Golden, "Optimization by simulated annealing: A preliminary computational study for tsp," presented at the N.I.H.E. Summer School on Combinatorial Optimization, 1983.
- [18] M. D. Levenson, N. S. Viswanathan, and R. A. Simpson, "Improving resolution in photolithography with a phase-shifting mask," *IEEE Trans. Electron Devices*, vol. ED-29, pp. 1828-1836, 1982.
- [19] S. K. Dumbrock, R. L. Lozes, A. Murray, C. Sauer, W. H. Arnold, D. Kyser, A. Minvielle, J. Nistler, M. Preil, B. Singh, and M. Templeton, "Phase-shift technology: Requirements for e-beam mask lithography," in *SPIE Symp. on Microlithography*, vol. 1264, 1991.
- [20] T. Terasawa, A. I. Hasegawa, T. Tanaka, and S. Katagiri, "Variable phase-shift mask for deep sub-micron optical lithography," in *SPIE Symp. on Microlithography*, vol. 1263, 1991.
- [21] Y. Liu, A. K. Pfau, and A. Zakhor, "Systematic design of phase-shifting masks with extended depth of focus and/or shifted focus plane," in *SPIE Symp. on Optical/Laser Microlithography*, vol. 1674-02, Mar. 1992.



Yong Liu (SM'91) was born in Yunnan province in China in 1962. He received the B.S. degree from Tsinghua University (China) in 1984 and M.S. degree from University of California at Berkeley in 1989, both in electrical engineering.

He is currently studying toward his Ph.D. degree in University of California at Berkeley in the department of electrical engineering and computer science. His current research interests include signal processing, microlithography, electromagnetics and microwave.



Avidesh Zakhor (S'87-M'87) received the B.S. degree from California Institute of Technology, Pasadena, and the S.M. and Ph.D. degrees from Massachusetts Institute of Technology, Cambridge, all in electrical engineering, in 1983, 1985, and 1987 respectively. Her doctoral thesis was on sampling and reconstruction schemes for multidimensional signals.

In 1988, she joined the Faculty at U.C. Berkeley where she is currently Assistant Professor in the Department of Electrical Engineering and Computer Sciences. Her research interests are in the general area of signal processing and its applications to images and video, and biomedical data. She has been a consultant to a number of industrial organizations in the areas of signal processing, communications and medical imaging, and has two pending patents on MRI signal processing, one on sigma delta modulators, and one on phase shifting mask design for optical lithography.

Ms. Zakhor was a General Motors scholar from 1982 to 1983, received the Henry Ford Engineering Award and Caltech Prize in 1983, was a Hertz fellow from 1984 to 1988, received the Presidential Young Investigators (PYI) award, IBM junior faculty development award, and Analog Devices junior faculty development award in 1990, and ONR young investigator award in 1991. She is a member of Tau Beta Pi, Sigma Xi, and the ASSP Society of the IEEE.