Journal of
Micro/Nanolithography,
MEMS, and MOEMS

Nanolithography.SPIEDigitalLibrary.org

Efficient three-dimensional resist profile-driven source mask optimization optical proximity correction based on Abbe-principal component analysis and Sylvester equation

Pei-Chun Lin
Chun-Chang Yu
Charlie Chung-Ping Chen

SPIE.

J. Micro/Nanolith. MEMS MOEMS 14(1), 011006 (Jan–Mar 2015)

Efficient three-dimensional resist profile-driven source mask optimization optical proximity correction based on Abbe-principal component analysis and Sylvester equation

Pei-Chun Lin,* Chun-Chang Yu, and Charlie Chung-Ping Chen
National Taiwan University, Graduate Institute of Electronics Engineering, No. 1, Sec. 4, Roosevelt Road, Taipei 10617, Taiwan

Abstract. As one of the critical stages of a very large scale integration fabrication process, postexposure bake (PEB) plays a crucial role in determining the final three-dimensional (3-D) profiles and lessening the standing wave effects. However, the full 3-D chemically amplified resist simulation is not widely adopted during the post-layout optimization due to the long run-time and huge memory usage. An efficient simulation method is proposed to simulate the PEB while considering standing wave effects and resolution enhancement techniques, such as source mask optimization and subresolution assist features based on the Sylvester equation and Abbe-principal component analysis method. Simulation results show that our algorithm is 20× faster than the conventional Gaussian convolution method. © *The Authors. Published by SPIE under a Creative Commons Attribution 3.0 Unported License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI.* [DOI: 10.1117/1.JMM.14.1.011006]

Keywords: Abbe-principal component analysis; Sylvester equations; postexposure bake diffusion and optical simulation.

Paper 14103SSP received Jun. 30, 2014; accepted for publication Nov. 7, 2014; published online Dec. 22, 2014.

1 Introduction

Photolithography is a critical process in modern integrated circuit manufacturing. As the technology advances, the feature size of the design pattern shrinks. Although the feature size of the design pattern is much smaller than the wavelength of the exposure light source, the optical diffraction effects and the acid diffusion effects of the chemically amplified resist become key factors in the printing fidelity.

Acid diffusion and the standing wave effect occur during the exposure which degrades the edge of the patterned image. The standing wave effect is shown in Fig. 1, which is generally solved by the postexposure bake (PEB) process. However, most of the simulations with resolution enhancement techniques (RETs) did not consider acid diffusion. Although some simulations include RETs, only a Gaussian function is applied to simulate acid diffusion. Although the influence of the acid diffusion during the PEB process increases with the scaling down of the line-width, we need to look into this issue more seriously.

In this paper, we consider the effect of the PEB process into the RETs. Unlike the traditional methods, the Sylvester equation is applied to simulate the PEB with an acid diffusion effect in a 20× faster runtime than the convolution with a negligible accuracy loss (10^{-16}). In addition, our simulation flow uses Abbe-principal component analysis (Abbe-PCA) to get the aerial image through the optical lens system,[1] which can execute source mask optimization (SMO) with a better performance of time efficiency than the Hopkins' method. We also propose an efficient method called image intensity error with an error penalty coefficient (EPC-IIE) to calculate the edge intensity error to get a fast judgment for every candidate image.

The rest of this paper is organized as follows. In Sec. 2, a new method for PEB acid diffusion based on the Sylvester equation will be proposed. Moreover, taking the proposed PEB acid diffusion solution into consideration, the RETs will be described in Sec. 3. The results are shown in Sec. 4.

2 PEB Acid Diffusion Simulation

PEB process, which is a chemical reaction, is an effective method of reducing standing waves. After the photoacid from the photoresist is generated, a chemical latent image will be formed through the photoacid diffusion. For faster and more accurate PEB process simulation, the Sylvester equation is applied to model the photoacid diffusion function.

2.1 *Postexposure Bake Diffusion*

In the exposure process, some portions of light are not absorbed by the photoresist and reach the substrate surface which reflects light. This produces constructive and destructive interference formed by the standing wave effect. To reduce the standing wave effect, the most useful method is the PEB[2] which is proposed by Walker.[3]

We use Fick's second law of diffusion to describe molecular diffusion. C_A is the concentration of species A, D_A is the diffusion coefficient of A at some PEB temperature T, t is the PEB time, and r represents different directions in different order.

$$\frac{\partial C_A}{\partial t} = D_A \frac{\partial^2 C_A}{\partial r^2}. \tag{1}$$

In general cases, it is accurate to assume that the diffusivity is independent of concentration. We can solve the

*Address all correspondence to: Pei-Chun Lin, E-mail: pejailu@gmail.com

diffusion equation of species A under general conditions and the initial distribution.

Now consider one possible initial condition known as the impulse source, which is also known as a delta function of concentration. For the boundary conditions, we assume zero concentration as r approaches $\pm\infty$; in Eq. (1). The solution of Eq. (1) is the Gaussian distribution, where σ is the diffusion length and r denotes the distance from the reference point to original point. In a general case, we can simulate the diffusion function by using a convolution method.

$$C_A(r) = \frac{N}{\sqrt{2\pi\sigma^2}} e^{-r^2/2\sigma^2}, \tag{2}$$

$$\sigma = \sqrt{2Dt}. \tag{3}$$

2.2 *Efficient Diffusion Equation Simulation Methods*

Now we consider the solution of the diffusion equation in one dimension, write the equation as $[\partial T(x,t)/\partial t] = D[\partial^2 T(x,t)/\partial x^2]$, and define its boundary condition as $x_l \le x \le x_h$. In the time domain, we discredited it into an equally spaced grid $t_n = t_n + n\delta t$, and let $T_i^n = T(x_i, t_n)$. We calculate the answer as in Eqs. (4) or (5) for each time step.

$$\frac{T_i^{n+1} - T_i^n}{\partial t} = D\frac{T_{i-1}^n - 2T_i^n + T_{i+1}^n}{\partial x^2}, \tag{4}$$

$$T_i^{n+1} = T_i^n - C(T_{i-1}^n - 2T_i^n + T_{i+1}^n), \quad C = D\frac{\partial t}{\partial x^2}. \tag{5}$$

Stability is the most important issue for numerical methods. If we set the wrong constraints the number will spread to infinity or be other under control states. Von Neumann stability analysis,[4] also known as Fourier stability analysis, is a procedure used to check the stability of finite difference schemes applied to linear partial differential equations. Consider the time evolution of a single-Fourier mode with wave number k:

$$\mathrm{T}\hat{}^{n+1} e^{ikx_n} = \mathrm{T}\hat{}^{n} e^{ikx_n}[1 + C(e^{-ik\delta x} - 2 + e^{ik\delta x})], \tag{6}$$

$$\mathrm{T}\hat{}^{n+1} = A\mathrm{T}\hat{}^{n}, \qquad A = 1 - 4C\sin^2\left(\frac{k\delta x}{2}\right). \tag{7}$$

Factor A is the amplitude of the Fourier mode at each time step. From Eq. (7), we can clearly see that the Fourier mode is stable when $C < (1/2)$.

2.3 *Crank–Nicolson Method*

The Crank–Nicolson method[5] is a finite difference method used for numerically solving the heat equation and similar partial differential equations. Equation (8) is the scheme that we use to revisit the original diffusion equation.

$$\frac{T(x,t_{n+1}) - T(x,t_n)}{\partial t} = \frac{D\partial^2 T(x,t_n)}{2\partial x^2} + \frac{D\partial^2 T(x,t_{n+1})}{2\partial x^2}. \tag{8}$$

We performed the von Neumann stability analysis for Eq. (8), with the amplified factor $A = [1 - 2C\sin^2(k\delta x/2)]/[1 + 2C\sin^2(k\delta x/2)]$. $|A| < 1$ for all values of k. It follows that the Crank–Nicolson scheme is unconditionally stable. The advantage of the Crank–Nicolson scheme is that not only is it unconditionally stable, but it can also be speeded up using matrix algebra.

Now, we consider the solution in two dimensions. By adopting the Crank–Nicolson scheme, we can get the iteration form as in Eq. (9):

$$\begin{aligned} T_{i,j}^{n+1} &- \frac{D\delta t}{2(\delta x)^2}(T_{i-1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i+1,j}^{n+1}) \\ &- \frac{D\delta t}{2(\delta y)^2}(T_{i,j-1}^{n+1} - 2T_{i,j}^{n+1} + T_{i,j+1}^{n+1}) \\ &= T_{i,j}^n + \frac{D\delta t}{2(\delta x)^2}(T_{i-1,j}^n - 2T_{i,j}^n + T_{i+1,j}^n) \\ &\quad + \frac{D\delta t}{2(\delta y)^2}(T_{i,j-1}^n - 2T_{i,j}^n + T_{i,j+1}^n). \end{aligned} \tag{9}$$

We set the number of grid points in the x and y directions as n and m, respectively. However, this is not the same as the number of time steps. We write it in matrix form then use matrix algebra to speed it up. We rewrite the equation in matrix form to calculate the solution for every grid point, where $I_x = \{(1/2) + [D\delta t/(\delta x)^2]\}I$ and $I_x = \{(1/2) + [D\delta t/(\delta y)^2]\}I$ are the identity matrix with the coefficient.

$$(M + I_y)T^{n+1} + T^{n+1}(N + I_x) = C, \tag{10}$$

$$M = \frac{-D\delta t}{2(\delta y)^2}\begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & \cdots & 0 & 0 & 0 \\ & & & & \vdots & & & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix},$$

$$N = \frac{-D\delta t}{2(\delta x)^2}\begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & \cdots & 0 & 0 & 0 \\ & & & & \vdots & & & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix}.$$

2.4 *Sylvester Equation*

Recalling Eq. (10), we rewrite our function to $AX + XB = C$, where $A \in R^{m\times m}$, $B \in R^{n\times n}$, and $X, C \in R^{m\times n}$, and we need to solve X. This equation is known as the Sylvester equation.

We have $m \times n$ equations to solve these $m \times n$ unknown numbers and no promise for yielding a unique solution. A proven fact is that matrix X has a unique solution if and only if the eigenvalues $\alpha_1, \alpha_2, \cdots \alpha_m$ of A and $\beta_1, \beta_2, \cdots \beta_n$ of B satisfy $\alpha_i + \beta_j \ne 0$.

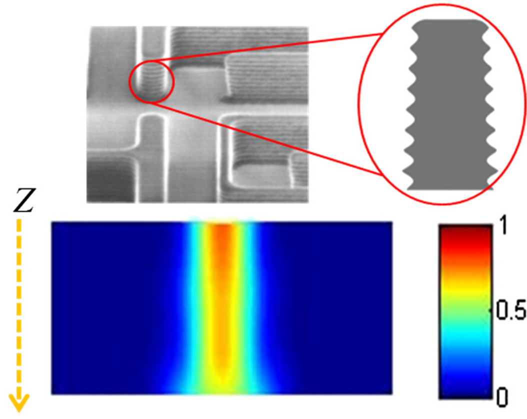

Fig. 1 The standing wave effect.

A faster method[6] to solve this equation is rewritten in the form $(I_m \otimes A + B^T \otimes I_n)\mathrm{vex}X = \mathrm{vex}C$, where I is the identity matrix and $\otimes$ is the Kronecker product notation and the vectorization operator vec. The Sylvester equation can be seen as a linear system of dimensions $mn \times mn$.

Recalling Eq. (10), matrices A and B are diagonally dominant, so the eigenvalues of matrices A and B are always larger than zero. Both of the matrices are positive definite and tridiagonal matrices. Therefore, our diffusion equation satisfies the definition of the Sylvester equation, and we can also use sparse matrix algorithms to speed up the performance or reduce the memory usage.

2.5 *3-D Formulation with Sylvester Equation*

During the PEB process, the correlation between different layers needs to be considered. There are tiny variations between the layers caused by not only the exposure process but also the PEB process. First, we cut the photoresist to k layers. The size of each is n by m. Equation (9) can be written as Eq. (11) in three-dimensions (3-D).

$$\begin{aligned} T^{n+1}_{i,j,l} &- \frac{D\delta t}{2(\delta x)^2}(T^{n+1}_{i-1,j,l} - 2T^{n+1}_{i,j,l} + T^{n+1}_{i+1,j,l}) - \frac{D\delta t}{2(\delta y)^2} \\ &(T^{n+1}_{i,j-1,l} - 2T^{n+1}_{i,j,l} + T^{n+1}_{i,j+1,l}) - \frac{D\delta t}{2(\delta z)^2} \\ &(T^{n+1}_{i,j,l-1} - 2T^{n+1}_{i,j,l} + T^{n+1}_{i,j,l+1}) = \\ &T^{n}_{i,j,l} + \frac{D\delta t}{2(\delta x)^2}(T^{n}_{i-1,j,l} - 2T^{n}_{i,j,l} + T^{n}_{i+1,j,l}) + \frac{D\delta t}{2(\delta y)^2} \\ &(T^{n}_{i,j-1,l} - 2T^{n}_{i,j,l} + T^{n}_{i,j+1,l}) \\ &+ \frac{D\delta t}{2(\delta z)^2}(T^{n}_{i,j,l-1} - 2T^{n}_{i,j,l} + T^{n}_{i,j,l+1}). \end{aligned} \tag{11}$$

We combine the computation of the z direction into the smaller size of A and B matrices. For example, according to the Sylvester equation $AX + XB = C$ as shown in Fig. 2, the dimensions of A, B, X, and C are now $A \in R^{m\times m}$, $B \in R^{(n\times k)\times(n\times k)}$, and $X, C \in R^{m\times(n\times k)}$. The size of B matrix changes from $n \times n$ to $(n \times k) \times (n \times k)$. Each layer of the main diagonal $\{\mathrm{Layer1}, \mathrm{Layer2}, \dots, \mathrm{Layer}k\}$ in Fig. 3 is a $(N + I_x) + [D\delta t/(\delta z)^2]I$ matrix as in Eq. (10). The upper block $\{U_{12}, U_{23}, \dots, U_{(k-1)k}\}$ and lower block

$$A_{m\times m}X_{m\times n} + X_{m\times n}B_{n\times n} = C_{m\times n}$$

A X + X B = C

Fig. 2 The diagram of the Sylvester equation in matrix form.

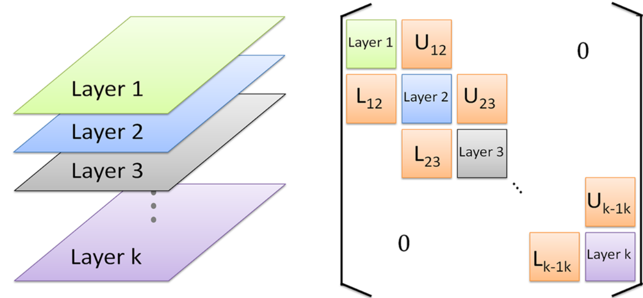

Fig. 3 The three-dimensional (3-D) formulation matrix.

$\{L_{12}, L_{23}, \dots, L_{(k-1)k}\}$ are the same size as the Layer. Every block $U_{(l-1)l}$ and $L_{(l-1)l}$ is a diagonal matrix with the value $-[D\delta t/2(\delta z)^2]$. As a definition, the new matrix is still a diagonally dominant matrix, so the matrix is still positive definite. We can use the Sylvester equation to solve it. The matrix is symmetric and sparse, so the space complexity is $O(n \times k)$. If we are in a uniform system, it means that the diffusion coefficient D is consistent in different places of the photoresist and we can save the memory usage as a constant.

3 Resolution Enhancement Techniques

With the feature size of devices shrinking and the number of transistors on the chip increasing, image simulation on the photoresist is facing the challenge to be both fast and accurate. Therefore, many resolution enhancement techniques have recently been proposed to improve the image resolution quality and make it easy for manufacture. In this work, we use the SMO and subresolution assist features (SRAFs) based on the Abbe-PCA to fast and accurately enhance the latent image quality.

3.1 *Source and Mask Optimization*

Many of the RETs mentioned above are in favor of improving the printing quality individually. There are two solutions for RETs; one is to optimize the light source, another one is to optimize the mask. SMO is the RETs that take not only the mask but also the source into consideration for an overall optimization. Therefore, SMO provides a larger solution space and leads to a better result than other simple RETs. In this work, we decide the source points in the first quarter, and then map them into the remaining three quarters. Therefore, we will generate four lookup tables every time when increasing or deleting a source point in the first quarter.

Initially, the image system is illuminated by the existing Abbe point sources of a given illuminator. As shown in Fig. 4, during source optimization, we first consider the candidate Abbe point sources outside the initial illuminator.

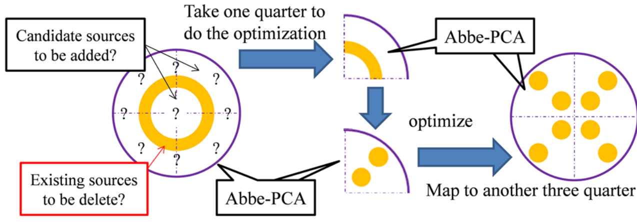

Fig. 4 Source optimization using Abbe-PCA.

Every candidate Abbe point source is considered to be added if the resultant cost function value is getting better. According to the properties of Abbe-PCA mentioned previously, Abbe-PCA kernel compaction can be performed repetitively after we add or delete any Abbe sources during the source optimization.

The mask optimization we use in this work is a combination of pixel-based optical proximity correction (OPC) methods. For our pixel-model-based OPC process, we denote the mask patterns during iterative mask optimization as $O = O_o + O_p - O_n$. As shown in Fig. 5, the notation that O_o is the initial mask pattern, which is also the target design pattern we expect, O_p is the candidate external pixels to be added, and O_n is the existing internal pixels to be deleted.

In our pixel-model-based OPC, the order of the searching sequence would significantly affect the final result and should be carefully decided. In the essence of optical and process proximity, nonideal effects are usually dominated by neighboring features. Therefore, we propose a breadth first search starting from the edges of the original mask pattern. The pixel-based OPC is performed gradually from the initial edges toward the insides and outsides. Moreover, the optimization masks are mostly chosen near the original mask

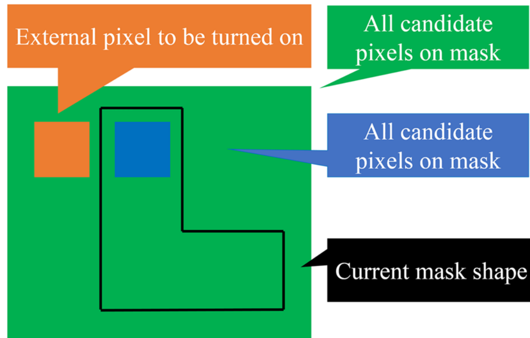

Fig. 5 Candidate internal and external pixels.

pattern. As a result, we use conventional model-based OPC first for the rough optimization with the large pixels. In addition, we perform pixel-based OPC for detailed optimization with the small-pixels on the previous model-based results hierarchically as shown in Fig. 6.

3.2 *Subresolution Assist Features*

SRAFs are another solution for RETs. Due to the interference effect, SRAFs is highly relative to λ_0/NA, where NA is the numerical aperture and λ_0 is the wavelength of the light source. The essence of the optical system, image resolution, is determined by the smallest lens which is also called the exit pupil with its diameter of numerical aperture. SRAFs with scattering bars have been proven effective in adjusting the critical dimension of isolated lines to match that of densely packed lines, and simultaneously improving the depth of focus of isolated lines to match the dense case. In this work, we place the assist features following the rule based on Eqs. (12)–(14), where the *MaxSrafNum* is the number of assist features we can add to every edge, the *SrafThick* is the width of the assist features in our work,[7] and the *SrafDistance* is the distance from one assist feature to another assist feature or the main feature. Under our calculation and experiment, we set *SrafThick* and *SrafDistance* as $0.19(\lambda_0/\text{NA})$ and $0.75(\lambda_0/\text{NA})$.

$$HitYLow = YLow - 2*MaxSrafNum*SrafThick - (2*MaxSrafNum+1)*SrafDistance, \quad (12)$$

$$HitXHi = XHi + MaxSrafNum*SrafThick + (MaxSrafNum+1)*SrafDistance, \quad (13)$$

$$HitXLow = XLow - 2*MaxSrafNum*SrafThick - (MaxSrafNum+1)*SrafDistance. \quad (14)$$

We generate the assist feature hit lists before we add the assist feature. The hit lists record the other rectangles that may overlap with the assist features of the tested rectangle. Taking the top edge of a tested rectangle as an example, we will inspect three blocks which will overlap with the assist features of the testing rectangle. First, we inspect the left block whose y coordinate is from *HitYLow* to *YLow*, and the x coordinate is from *HitXLow* to *XLow*. Second, we inspect the right block whose y coordinate is from *HitYLow* to *YLow*, and the x coordinate is from *XHi* to *HitXHi*. In the end, we inspect the central block whose y

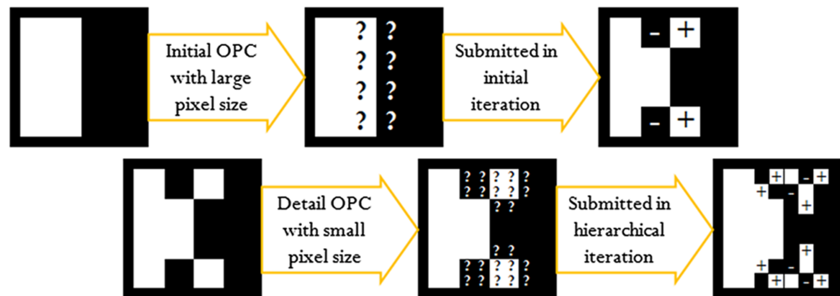

Fig. 6 Hierarchical pixel-model-based OPC.

coordinate is from *HitYLow* to *YLow*, and the x coordinate is from *XLow* to *XHi*. Notations that the *XLow*, *XHi*, *YLow* are the boundaries of the tested rectangle, and *HitYLow*, *HitXHi*, *HitXLow* are indicated in Eqs. (12)–(14).

3.3 *Partial Coherent Illumination*

Describing the partial coherent illumination, the mutual intensity $J(x,y)$ is applied to the image formulation. $J(x,y)$ can be derived from Fourier transformation to the physical shape of illumination $\tilde{J}(x,y)$. The symbol σ in Eq. (15) is defined as the diameter ratio between the illuminator and the exit pupil, which is equivalent to the essence of the optical lens system of the lithography.

$$\tilde{J}(f,g)=\begin{cases}1, & \sqrt{f^2+g^2}<\sigma\\ 0, & \text{otherwise}\end{cases}. \tag{15}$$

3.4 *Abbe's Image Formulation*

For the partial coherent system, there are two approaches for imaging, one is Hopkins' theory[8] and another is Abbe's method. Hopkins' imaging requires the transmission cross-correlation (TCC) matrix of the illuminating pattern with the pupil and its complex conjugate. To generate the TCC matrix, a space complexity of $O(n^4)$ is needed. Therefore, we choose Abbe's method to formulate the image system.

$$\begin{aligned}I(x,y:z)=I_{\max}^{-1}&\iiint\iiint \tilde{J}(f,g)\tilde{H}_0(f+f',g+g':z)\\ &\tilde{O}(f',g')\cdot\tilde{H}_0^*(f+f'',g+g'':z)\\ &\tilde{O}^*(f'',g'')e^{-i2\pi[(f'-f'')x+(g'-g'')y]}\mathrm{d}f\,\mathrm{d}g\,\mathrm{d}f'\,\mathrm{d}g'\,\mathrm{d}f''\,\mathrm{d}g'',\end{aligned} \tag{16}$$

$$\begin{aligned}I(x,y:z)=I_{\max}^{-1}\iint\tilde{J}(f,g)\bigg|\iint\tilde{H}_0(f+f',\\ g+g':z)\tilde{O}(f',g')e^{-i2\pi(f'x+g'y)}\mathrm{d}f'\,\mathrm{d}g'\bigg|^2\mathrm{d}f\,\mathrm{d}g,\end{aligned} \tag{17}$$

$$\begin{aligned}I(x,y:z)=I_{\max}^{-1}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\tilde{J}(f,g)\\ |H_0(x,y:z)e^{-i2\pi(fx+gy)}*O(x,y)|^2\mathrm{d}f\,\mathrm{d}g.\end{aligned} \tag{18}$$

Abbe's image formulation is based on approximating the illuminator into a sum of discrete point sources $\tilde{J}(f,g)\cong\sum_s^{\text{source}}a_s\delta(f-f_s,g-g_s)$. We rewrite the Hopkins' formulation as shown in Eq. (16) into a squared integration of convolution results with respect to the physical shape of the illuminator. Equation (18) shows that the image intensity can be calculated by summing the square of the light field caused by each Abbe source.

3.5 *Abbe-Principal Component Analysis Method*

PCA is mathematically defined as an orthonormal linear transformation. PCA transforms the data to a coordinate spanned by its eigenvectors such as the greatest eigenvalue. In addition, PCA can be used for dimensionality reduction in a data set by retaining the principal components which contain the greater eigenvalue and neglecting the nonprincipal components. However, most of its characteristics still remain.

We modify Eq. (18) to Eq. (19), where a_s is the relative intensity of each Abbe point source. Our goal is to rebuild the coherent systems by using principal eigenvectors of the image kernel space to approximate the behavior of the sum of the coherent system built by Abbe's method.[1]

$$\begin{aligned}I(x,y:z)&=I_{\max}^{-1}\sum_s^{\text{source}}a_s|H_s(x,y:z)*O(x,y)|^2\\ &=I_{\max}^{-1}\sum_c^{\text{kernel}}\lambda_c|\varphi_c(x,y:z)*O(x,y)|^2,\end{aligned} \tag{19}$$

$$HH^*=(USV^*)(USV^*)^*=USV^*VS^*U^*=US^2U^*. \tag{20}$$

We describe the image system as Eq. (19). We suppose there are k Abbe point sources, and we can generate k kernels with size n by n. We can set the kernel into a matrix H with dimension n^2 by k. We use the singular value decomposition method, and derive it as $H=USV^*$, where U is the eigenvectors of the image kernel space, V is the coefficients of the point spread function linear combination, and S is the singular values. We can rewrite the image system as $I(x,y:z)=I_{\max}^{-1}\sum|O(x,y)*US|^2$ by Eq. (20). Thus, the PCA transformation can be written as $K=HV=US$, which means the PCA space K can be spanned by a linear combination of PCA kernels ϕ_c in U with the weights of singular values in S. As shown in Fig. 7, using Abbe-PCA reduces the kernel size from $n^2\times k$ to $n^2\times k'$.

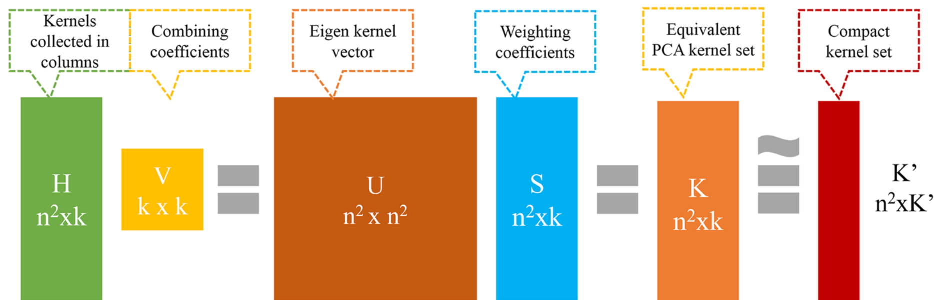

Fig. 7 Abbe-PCA compaction.

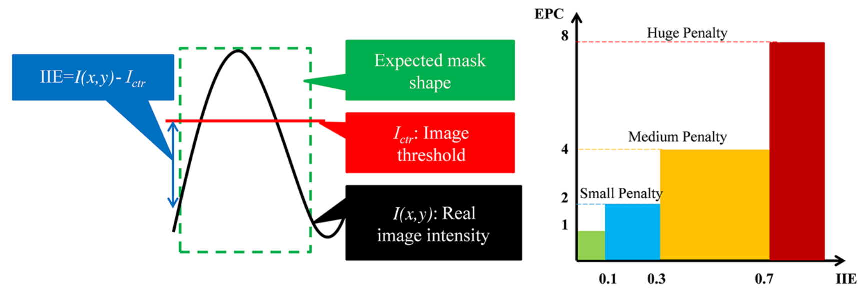

Fig. 8 Error penalty coefficient.

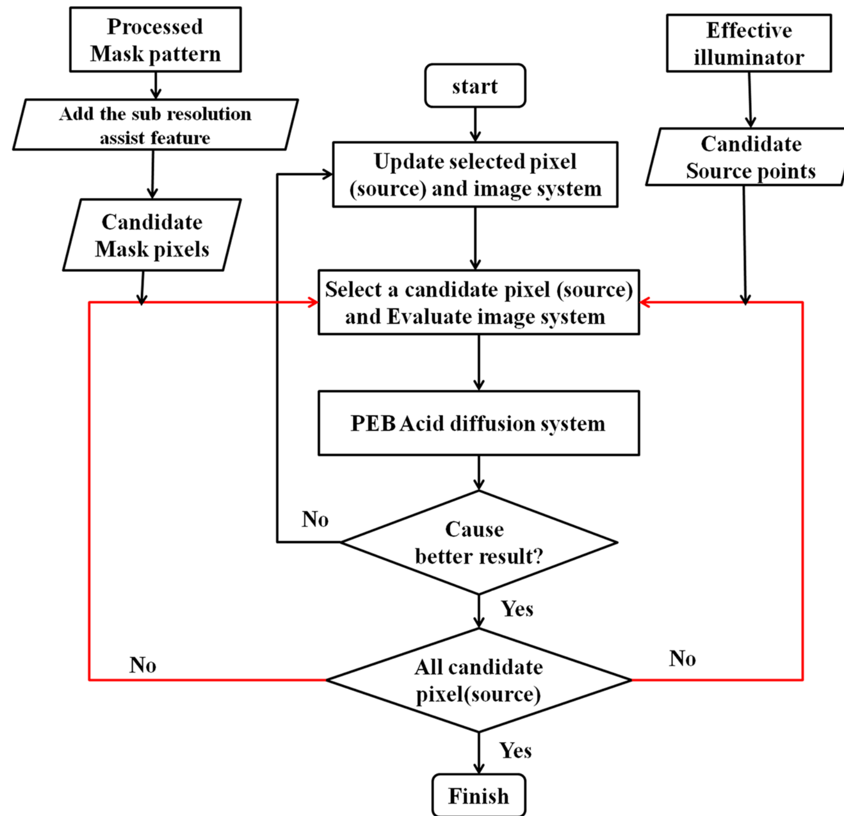

Fig. 9 Our flowchart of resist image simulation.

3.6 *Error Penalty Coefficient*

Traditionally, the cost function used in RETs is measured by the edge placement error (EPE),[9] which calculates the difference between the image contour and the excepted design pattern. However, EPE is time-consuming since it involves the full image simulation demanded and is hard to handle with a mathematical expression. In this work, we propose another cost function called EPC-IIE. The EPC-IIE is described in Eq. (21). Notation $O_{e,\mathrm{ctr}}(x,y)$ means the sampling function at the edges of the whole design pattern, and I_{ctr} means the contour threshold for a constant image model. EPC-IIE is defined as generalized edge intensity error with extra two contours and image thresholds, which does not only optimize the edge intensities but also their adjacent two contours with different thresholds. The definition of the error penalty coefficient is like a piecewise linear step function, which has a different constant coefficient in every image intensity error range as illustrated in Fig. 8.

Table 1 Optical parameters of lithography system for 45-nm immersion.

Description of parameter	Notation	Value
Image space refractive index	n_{img}	1.45
Maximum incident angle	$\theta_{\max}$	1.25 rad
Numerical aperture	NA	1.35
Exposure wavelength	λ_0	193 nm
Defocus	d_{defocus}	0.0
Diffusion coefficient	D	3.44×10^{-20} m^2 s^{-1} K^{-1}
Postexposure bake (PEB) time	t	50 s
PEB temperature	T	150°C

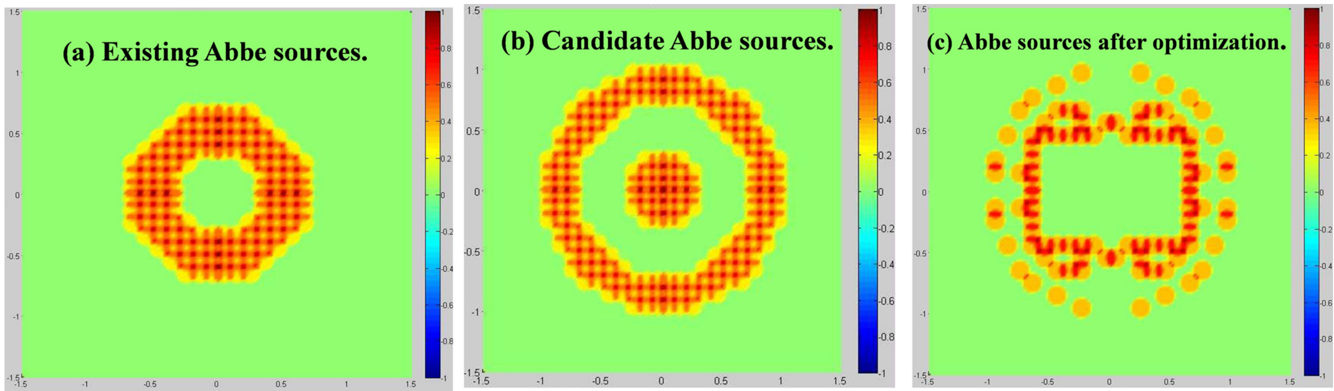

Fig. 10 (a) The existing Abbe sources, (b) the candidate Abbe sources, and (c) Abbe sources after optimization.

Fig. 11 (a) The original mask, (b) the original aerial image, (c) the image after postexposure bake (PEB), (d) difference between (b) and (c), (e) the image contour of (b), (f) the image contour of (d), (g) the mask after source mask optimization and subresolution assist features, and (h) the final effective latent image.

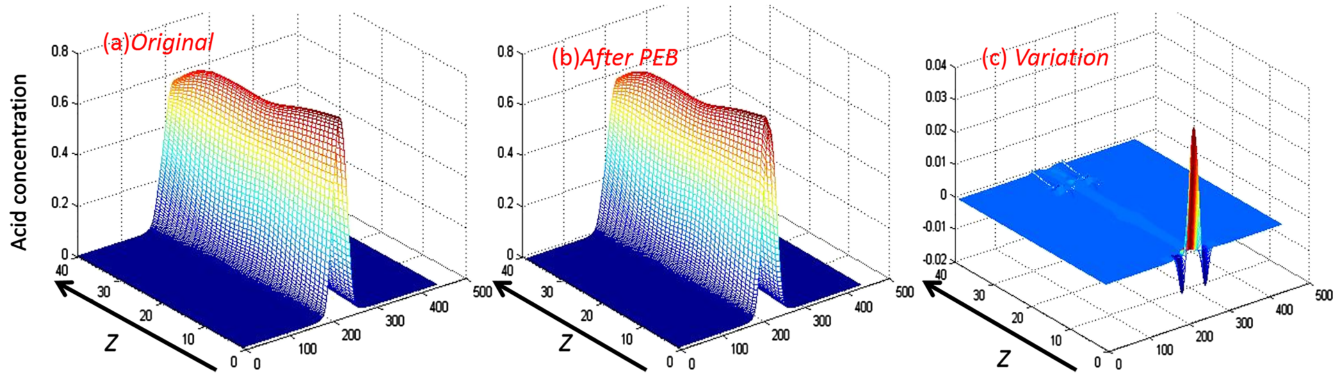

Fig. 12 (a) The original acid concentration, (b) the acid concentration after PEB, and (c) the variation of the acid concentration.

Table 2 The average image intensity error with an error penalty coefficient (EPC-IIE) for all edge points in Fig. 11.

Status	Original	Subresolution assist feature (SRAF)	Source optimization + SRAF	Source mask optimization + SRAF
Average EPC-IIE cost	0.0708	0.0705	0.0577	0.0314
Reduced	—	1.87%	18.52%	55.73%

$$\mathrm{EPC-IIE} = \sum_{\mathrm{ctr}}^{\mathrm{contour}} \sum_{e}^{\mathrm{edge}} P(\mathrm{IIE})|I(x,y) - I_{\mathrm{ctr}}|O_{e,\mathrm{ctr}}(x,y). \tag{21}$$

4 Simulation Result

We contruct our system in MATLAB 2010, following the simulation flowchart shown in Fig. 9. As for our hardware environment, the CPU is Intel Core i7-2600 3.1 GHz with 16-GB RAM. Windows 7 is also applied as our operating system.

We use the typical 45-nm immersion lithography[10] in our work. We also use a mesh point up to 20 for the accuracy of kernels, namely the exit pupil in the spectrum domain is a circ function with a radius up to 10. In addition, we downscale the mask and kernels with a scaling factor 0.6, so that the simulation resolution is under 1.7 nm which saves runtime, and the diffusion coefficient is $3.44 \times 10^{-20}\ \mathrm{m^2\,s^{-1}\,K^{-1}}$.[11] The detailed parameters are listed in Table 1, and the thickness of the photoresist is 100 nm.

Moreover, in the PEB process, our method demonstrates a speed up around 20× over the convolution method for a single time step (2 s) as illustrated in Fig. 11(b). To simulate a 50-s flexible PEB diffusion process, our method only takes 250 s with a convolution set up for a 1860×1860 working area. To complete the resist image from the exposure process, it takes less than 120 s for each iteration, with a specified iteration cycle.

Figure 10 shows the original existing Abbe sources, the candidate Abbe sources and the final sources after optimized the design pattern as shown in Fig. 11(a).

Figure 11(a) shows the original mask and Fig. 11(g) shows the optimized mask after our flowchart. As shown in Figs. 11(d)–11(f), there are significant differences between the aerial image and the image after PEB because both the image intensity and the contour shape of them are different. Figure 12 shows the acid concentration before and after PEB along the z direction, which represents the acid concentration at the short edge of the polygon X in Fig. 11(b). The scaling factor of the z direction is 0.4 for the photoresist thickness, so the thickness of the photoresist is 100 nm. As shown in Fig. 12(a), we can clearly see the standing wave. The variation of the acid concentration after the PEB is shown in Fig. 12(c). We can see the acid diffuse from higher concentration ($z = 0$) to lower concentration ($z = 40$). We set a value 0.3 for the image threshold to calculate the EPC-IIE with the acid diffusion effect. In Eq. (21), we realize the latent image is more similar to the original mask if the EPC-IIE is small. The average EPC-IIE for all edge points is shown in Table 2, which shows the RETs we implemented.

5 Conclusions

In this paper, we consider the acid diffusion effect in our in-house simulator. In addition, the Sylvester equation is applied to simulate the PEB acid diffusion. Therefore, when simulating the PEB acid diffusion with a 3-D Sylvester equation, the simulation time is 20× faster than convolution method with a negligible accuracy loss (10^{-16}). Moreover, our result will be closer to the real shape after the processes.

Acknowledgments

The authors would like to thank TSMC nPtD/MEB to support this research.

References

1. J. H.-C. Chang, C. C.-P. Chen, and L. S. Melvin III, "Abbe-PCA-SMO: microlithography source and mask optimization based on Abbe-PCA," *Proc. SPIE* **7640**, 764026 (2010).
2. C. A. Mack, *Fundamental Principles of Optical Lithography: The Science of Microfabrication*, John Wiley & Sons, Hong Kong (2007).
3. E. J. Walker, "Reduction of photoresist standing-wave effects by post-exposure bake," *IEEE Trans. Electron Devices* **22**, 464–466 (1975).
4. J. G. Charney, R. Fjørtoft, and J. von Neumann, "Numerical integration of the barotropic vorticity equation," *Tellus* **2**, 237–254 (1950).
5. J. Crank and P. Nicolson, "A practical method for numerical evaluation of solutions of partial differential equations of the heat conduction type," *Proc. Camb. Philos. Soc.* **43**, 207–226 (1947).
6. A. Jameson, "Solution of equation $AX + XB = C$ by inversion of an $M \times M$ or $N \times N$ matrix," *SIAM J. Appl. Math.* **16**(5), 1020–1023 (1968).

7. S. M. Mansfield et al., "Semiconductor device fabrication using a photomask with assist features," US Patent No. 6,421,820 B1 (2002).
8. H. H. Hopkins, "The concept of partial coherence in optics," *Proc. Royal Society A* **208**(1093), 263–277 (1952).
9. N. B. Cobb, "Fast optical and process proximity correction algorithms for integrated circuit manufacturing," Doctor Dissertation, University of California, Berkeley (1998).
10. A. K. K. Wong, *Optical Imaging in Projection Micro-lithography*, SPIE Press, Bellingham, WA (2005).
11. J.-B. Park et al., "Acid diffusion length corresponding to post exposure bake time and temperature," *Jpn. J. Appl. Phys.* **46**(1), 28–30 (2007).

Pei-Chun Lin is pursuing his PhD degree at National Taiwan University. He received his BS degree in mathematics from National Cheng Kung University in 2010 and his MS degree in electrical engineering from National Taiwan University in 2012. His research interests include next-generation lithography, image processing, design for manufacturing, data processing, and the telecare system.

Chun-Chang Yu is pursuing his PhD degree at National Taiwan University. He received his BS degree in mechanical engineering from National Chung Cheng University in 2004, and his MS degree in electrical engineering from National Cheng Kung University in 2006. He has also worked for Andes Technology.

Charlie Chung-Ping Chen received his BS degree in computer science and information engineering from the National Chiao-Tung University, Hsinchu, Taiwan, in 1990 and his MS and PhD degrees in computer science from the University of Texas at Austin in 1996 and 1998. He is a professor in the EE Department of National Taiwan University, Taiwan. His research interests are in the areas of computer-aided design and microprocessor circuit design with an emphasis on interconnect and circuit optimization, circuit simulation, design for manufacturing, and signal/power/thermal integrity analysis and optimization.