Illumination source optimization in optical lithography via derivative-free optimization

Wen Lv,[1,2] Shiyuan Liu,[1,*] Xiaofei Wu,[2] and Edmund Y. Lam[2,3]

[1]*State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan 430074, China*
[2]*Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam, Hong Kong, China*
[3]*e-mail: elam@eee.hku.hk*
**Corresponding author: shyliu@mail.hust.edu.cn*

Received July 7, 2014; revised September 24, 2014; accepted September 29, 2014; posted October 3, 2014 (Doc. ID 216384); published November 3, 2014

Illumination source optimization (SO) in optical lithography is generally performed under a simulation model that does not consider critical effects such as the vectorial nature of light and mask topography. When a numerical aperture becomes large and the critical dimension reaches subwavelength, the prediction of this model generally fails; therefore, the previous works based on this model become inaccurate. In order to correctly compute SO, we first propose a new source pattern representation method that has moderate parameter variations but remains complete in solution space. Then we develop a derivative-free optimization (DFO) method to optimize these parameters under a rigorous simulation model. Unlike gradient-based techniques, DFO methods do not require a closed-form formulation of the model and are independent of the form of cost function.

OCIS codes: (110.5220) Photolithography; (110.1758) Computational imaging; (110.4235) Nanolithography.
http://dx.doi.org/10.1364/JOSAA.31.000B19

1. INTRODUCTION

Illumination source is a vital component in optical lithography and has attracted great interest due to its capacity for improving lithography resolution and imaging performance [1–3]. Mathematical optimization methods are popularly used to design the source, which is commonly referred to as source optimization (SO). Generally, SO is performed for specific mask patterns [4–6] or serves as a key part in the simultaneous source and mask optimization [7–10]. In this paper, we focus on the former.

Generally, a successful SO mainly relies on three aspects: source representation, cost function, and solution method [11]. Light is an electromagnetic wave consisting of amplitude, phase, and polarization state. Phase modulation can be conducted by pupil domain [12,13], and polarization state can be controlled by additional elements [14]. In this paper, we focus our effort on source pattern design, i.e., light intensity and position. Source patterns in optical lithography have experienced a series of evolutions. In the beginning, the illumination hardware had only two to three parameter variations (for example, inner radius, outer radius, and open angle), so source patterns were circular, annular, dipole, quadrupole, etc. With the advent of a diffractive optical element (DOE) and programmable illuminator (PI), more degrees of freedom on position and intensity of the light were provided [15,16]. They essentially pixelated the illumination control. So the pixel-based representation methods, which discretize source pattern as a raster or polar image constituted by pixels, became possible [8,17–19]. However, this flexibility, in turn, leads to significant cost in the freedom of optimization variables and runtime. To lessen the computational burden, contour-based representation methods (such as the level-set method) were developed [20], but these did not fully exploit the flexibility of PI and DOE in source intensity. Recently, a kernel-based representation method was proposed [21,22]. This method considers some well-chosen Zernike polynomials or 2D discrete cosine functions as basis functions, and the source pattern is decomposed into the linearly weighted superposition of these basis functions. As a result, the number of source parameters decreases significantly. However, these basis functions usually contain negative values and, therefore, potentially produce a negative source, which is unavailable with the current illumination hardware. So extra constraints must be added in SO to ensure the regularity of the computed source with these basis functions.

Cost function plays a big role in SO. Considering a simplified model without taking into account some critical effects such as the vectorial nature of light and mask topography, the output pattern under a specific source has a closed-form formulation [23]. In order to analytically calculate the gradient (sensitivity or derivative) of the cost function with respect to source parameter variations, closed-form metrics such as image fidelity and edge distance error are introduced to reveal the critical dimension (CD) error information on average, and image contrast, to act as the normalization image log slope (NILS) [18,24,25]. As a result, it achieves significant improvement in computational efficiency. However, these metrics encounter two main drawbacks. One is that they are substitutes for practical metrics (such as CD error and NILS); the other is that the closed-form formulation of an accurate model is sometimes complicated or even unavailable [19,26]. That

means these metrics would fall short when performing SO under an accurate model.

As the numerical aperture becomes large (for example, 1.35 in immersion lithography) and CD reaches subwavelength, the previous scalar thin-mask model became insufficient [23,26]. To maintain the accuracy of the model, the vectorial nature of light and mask topography must be considered, which leads to a complicated model. Therefore, it is difficult to derive a closed-form formulation of the SO problem under this accurate model. Analytical formulation of the gradient is even unavailable. Most recently, Ma and coauthors extend SO to a vectorial model continuing to use gradient-based methods but without taking rigorous mask models [22,27]. On the other side, Fühner and coauthors propose a genetic algorithm (GA) to solve SO [13,19,28]. GA is a metaheuristic method and does not need gradient information but enough iteration to obtain an optimal solution. It is time-consuming to optimize a source with a lot of parameters.

In this paper, we first propose an improved kernel-based source pattern representation method by adding an operator to the linear superposition of the basis functions. Section 2 details this method. Then we formulate the SO problem by using common metrics instead of the substitutes to evaluate the source performance in Section 3. Finally, we introduce a derivative-free optimization (DFO) method to handle SO in Section 4. DFO methods are recently proposed and have shown great promises to handle "black box" optimization problems whose gradient calculation is unavailable and cost function evaluation is expensive [29,30]. In this section, we develop a suitable DFO method to perform SO based on the source representation method. Section 5 provides the simulation results to demonstrate the validity of the proposed method, and we draw some conclusions in Section 6.

2. SOURCE PATTERN REPRESENTATION

Source intensity is a 2D, bounded, real-valued function inside a circle of radius σ (partial coherence factor):

$$0 \le s(f,g) \le s_{\max}, \qquad f^2+g^2 \le \sigma^2. \tag{1}$$

Here, f and g are spatial frequency coordinates, and $s_{\max}$ is a design tolerance to avoid sharp spikes damaging the projection lenses [11]. For simplicity, we normalize the tolerance $s_{\max}$ to 1 and the radius σ to 1. So the source intensity is

$$0 \le s(f,g) \le 1, \qquad f^2+g^2 \le 1. \tag{2}$$

With these constraints, we represent the source pattern as

$$s(f,g) = s([c_0,c_1,\ldots,c_n]) = T\left\{\sum_{i=0}^{n} c_i\omega_i(f,g)\right\}, \tag{3}$$

where $\omega_i(f,g)$ is called the kernel function or basis function, c_i is its corresponding coefficient, and $T\{\cdot\}$ is an operator to adjust source intensity to satisfy the requirements in Eq. (2). For simplicity, we denote the kernel coefficients c_i in Eq. (3) as a coefficient vector $\mathbf{c}$:

$$\mathbf{c} = [c_0,c_1,\ldots,c_n]. \tag{4}$$

Thus, the source pattern representation is changed from the pixelated spatial frequency coordinates (f,g) to a coefficient vector $\mathbf{c}$. As a result, source parameter variables decrease significantly. It is also noted that the value of $\omega_i(f,g)$ usually ranges from $-\infty$ to $+\infty$, so $T\{\cdot\}$ should be a map from $(-\infty,+\infty)$ to [0,1]. The following shows two examples of $T\{\cdot\}$:

$$T\{x\} = \frac{1+\cos(x)}{2}\cdot D(f,g), \tag{5}$$

$$T\{x\} = \begin{cases} 0, & x<0 \\ x\cdot D(f,g), & 0\le x\le 1 \\ D(f,g), & x>1 \end{cases}. \tag{6}$$

Here, $D(f,g)$ is unit disk function; the value inside the circle is 1 and outside is 0. Equation (5) is also commonly used in mask representation [31].

Regarding the choice of basis functions $\omega_i(f,g)$ in Eq. (3), any complete and orthogonal space under a certain inner product can be adopted, such as Zernike polynomials, circle sampling functions [32], and 2D discrete cosine functions. Here, we choose Zernike polynomials as raw basis functions. Since Zernike polynomials only have value on the unit disk, the operator $D(f,g)$ in Eqs. (5) and (6) can be ignored.

It is well-known that Zernike polynomials span the complete space on the unit disk. The complete Zernike polynomials are redundant on expressing the source pattern out of the symmetry of source. In other words, the coefficient of some Zernike polynomials can be set to zero.

Based on Eq. (3), the kernel coefficient c_i can be calculated as

$$c_i = \frac{\iint_{-\infty}^{\infty} T^{-1}\{s(f,g)\}\omega_i(f,g)\mathrm{d}f\mathrm{d}g}{\iint_{-\infty}^{\infty}\omega_i^2(f,g)\mathrm{d}f\mathrm{d}g}. \tag{7}$$

Note that most mask patterns in optical lithography are Manhattan; in order to avoid nontelecentricity caused by oblique incident, the source pattern is, therefore, f axis and g axis, even symmetrical [23]. Based on the symmetry, we conclude that the coefficient c_i is 0 when its corresponding Zernike polynomial is odd symmetrical on either f axis or g axis. Thus, getting rid of the Zernike polynomials that are odd symmetrical on either f axis or g axis, the rest are chosen, and

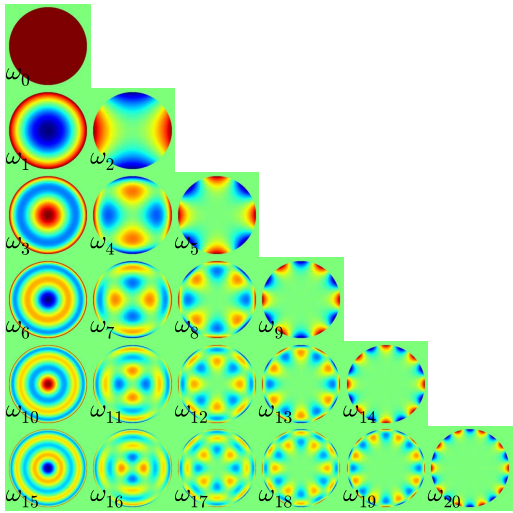

Fig. 1. First 21 basis functions selected from Zernike polynomials.

they remain complete to express the source pattern. In this paper, we normalize the maximum of the absolute value of each basis function to 1 and set the truncation order n in Eq. (3) to 54. Figure 1 shows the first 21 basis functions.

3. COST FUNCTION DEFINITION AND SOURCE OPTIMIZATION PROBLEM FORMULATION

The design of a proper cost function is crucial for SO. Generally, the cost function consists of various metrics used to evaluate the source performance for specific purposes.

Critical dimension (CD) error along different positions is a common metric to evaluate the difference between the printed pattern and the desired pattern, and its robustness with respect to focus fluctuation in manufacturing process must be considered. Here, we define a CD error metric as

$$O_{\mathrm{CD}}(\mathbf{c}) = \sum_{p=1}^{P} \mu_p [\mathrm{CD}(\mathbf{c}, h; p) - \mathrm{CD}^*(p)]^2, \tag{8}$$

where $\mathbf{c}$ is the coefficient vector of the source defined in Eqs. (3) and (4), h is defocus, p represents metrology position, μ_p is the corresponding weight for different position, $\mathrm{CD}(\mathbf{c}, h; p)$ represents the measured CD under the illumination source $s(\mathbf{c})$ and defocus h at the metrology position p, and $\mathrm{CD}^*(p)$ represents the desired CD at the position p. References [11,33] point out that the optimization results under out-of-focus are almost the same but without increasing runtime compared with that under an averaging performance through focus.

The normalized image log slope (NILS) is used to evaluate the process sensitivity to dose variation [1]. A higher NILS value is preferred. In order to be expressed as a minimization problem, the NILS metric is formulated as

$$O_{\mathrm{NILS}}(\mathbf{c}) = \sum_{p=1}^{P} \eta_p \left[\frac{\tau_p}{\mathrm{NILS}(\mathbf{c}; p)}\right]^2, \tag{9}$$

where τ_p is a positive number to control the sensitivity of the metric $O_{\mathrm{NILS}}(\mathbf{c})$ against the NILS changes; $\mathrm{NILS}(\mathbf{c}; p)$ denotes the computed NILS under the illumination source $s(\mathbf{c})$ at the metrology position p, and η_p is the corresponding weight.

To evaluate the quality of source patterns in smoothness (i.e., avoid sharp peaks), we define the source quality metric as

$$O_{\mathrm{S}}(\mathbf{c}) = \iint_{-\infty}^{\infty} \left[\frac{s(\mathbf{c})}{\iint_{-\infty}^{\infty} s(\mathbf{c}) \mathrm{d}f \mathrm{d}g}\right]^2 \mathrm{d}f \mathrm{d}g = \frac{\|s(\mathbf{c})\|_2^2}{\|s(\mathbf{c})\|_1^2}, \tag{10}$$

where $\|\cdot\|_1$ and $\|\cdot\|_2$ represent ℓ_1 and ℓ_2 norm, respectively.

It is noted that three metrics need to be minimized, i.e., CD error metric, NILS metric, and source quality metric. Generally, they are combined with certain proportions α, β, and γ, i.e.,

$$B(\mathbf{c}) = \alpha \cdot O_{\mathrm{CD}}(\mathbf{c}) + \beta \cdot O_{\mathrm{NILS}}(\mathbf{c}) + \gamma \cdot O_{\mathrm{S}}(\mathbf{c}). \tag{11}$$

$B(\mathbf{c})$ is the overall cost function. The SO problem is formulated as finding a coefficient vector $\mathbf{c}$ to minimize the cost function $B(\mathbf{c})$ as

$$\mathbf{c}^* = \arg\min_{\mathbf{c}} B(\mathbf{c}). \tag{12}$$

4. DERIVATIVE-FREE OPTIMIZATION METHOD

From Eq. (11), the cost function $B(\mathbf{c})$ contains the CD and NILS evaluation procedure under an illumination source pattern represented via coefficient vector $\mathbf{c}$. Its analytical gradient (sensitivity or derivative) with respect to coefficient vector $\mathbf{c}$ is usually complicated or even unavailable, and estimating the gradient by numerical differentiation methods may be prohibitively costly. In addition, the estimated gradient by numerical differentiation methods may be inaccurate if the cost function evaluation is noisy. To address these problems, DFO methods are proposed [29,30]. DFO methods, as they are called, do not require derivatives information and have shown great promise in handling "black box" optimization problems whose gradient is unavailable and function evaluation is expensive and noisy. Here, the cost function $B(\mathbf{c})$ is treated as a "black box", and we make use of one of these DFO methods, named the interpolation-based trust-region method, which is a combination of a trust-region framework with quadratic interpolation of the cost function [29,30]. Essentially, this method approximates the cost function itself instead of its gradient.

One distinction of our proposed method from the previous methods in [29,30] is the approximation and optimization strategy for variables in cost function. Let us assume that the cost function depends on n variables. The previous methods directly approximate the cost function with quadratic polynomials. For a full quadratic interpolation, including cross terms of each variable, it needs $(n+1)(n+2)/2$ sampling points inside a trust region (also means taking the same number of cost function evaluations), referring to Chap. 3 in [30]. Unfortunately, it requires an extra procedure to ensure well-poisedness of the sampling points and needs careful exploration on setting the trust-region range. If the interpolation fails to approximate the cost function, it needs to reset the region range, reselect the sampling points, and recalculate the time-consuming cost function evaluations.

While our proposed method considers a single variation in the cost function at a time, it is much easier to approximate and find a descent value. Moreover, the trust-region range can be more aggressive to accelerate convergence. In addition, our method is well suited for the proposed source pattern representation method. The descent rate and trust-region range of cost function with respect to each variable can be conveniently studied. The following presents the details of our proposed method.

In order to solve Eq. (12), we first consider a single coefficient c_i in coefficient vector $\mathbf{c}$ with the others fixed and define a trust region as $[c_i - \Delta, c_i + \Delta]$, where Δ is called trust-region radius. Then we formulate a subproblem as

$$c_i^* = \arg \min_{\substack{x_i \in \\ [c_i - \Delta, c_i + \Delta]}} B([\ldots, x_i, \ldots]). \tag{13}$$

To solve this problem, we approximate $B([\ldots, x_i, \ldots])$ with a quadratic polynomial in the region:

$$B([\ldots,x_i,\ldots]) \approx Q(x_i) = a_2^2 x_i^2 + a_1 x_i + a_0,$$
$$x_i \in [c_i - \Delta, c_i + \Delta]. \tag{14}$$

Here, a_2, a_1, and a_0 are, respectively, the coefficients of the quadratic, linear, and constant terms. The $Q(x_i)$ is computed via three sampling points, which are, respectively, $(c_i - \Delta, B([\ldots, c_i - \Delta, \ldots]))$, $(c_i, B([\ldots, c_i, \ldots]))$, and $(c_i + \Delta, B([\ldots, c_i + \Delta, \ldots]))$. Since $Q(x_i)$ is a quadratic function, we can easily find its minimum point c_i° in the region with three known points. Referring to the quadratic function $Q(x_i)$, the main principles to compute the minimum of $B([\ldots,x_i,\ldots])$ are as follows:

1. Determine the minimum: (1) The vertex of the quadratic function is either not contained in the trust region or it is a maximum. The minimum is given by either the left or the right bound of the trust region; (2) the vertex is contained in the trust region, and it is a minimum. Then the coordinate of the vertex is determined.
2. Model control and trust-region adjustment: If, in the case 1.(2), the actual function value corresponding to the minimum point of the interpolation function is worse than the initial value $B([\ldots, c_i, \ldots])$, refine the model by halving the trust region in radius. As an additional termination criterion, the trust region may not become smaller than a predefined minimum radius.

The corresponding pseudocode of the algorithm is detailed in Table 1.

From Table 1, we use the condition $B([\ldots, c_i^\circ, \ldots]) \le b_2$ to judge whether the trust-region radius is acceptable or not, instead of the commonly used metrics such as $\{b_1 - B([\ldots, c_i^\circ, \ldots])\}/\{b_1 - Q(c_i^\circ)\}$ [29,30]. This is because we are concerned with the descent of the cost function more than the approximation accuracy in this specific case. In a strict sense, the optimized coefficient c_i^* is not the exact optimal solution, which would be too expensive to compute. The $\Delta_{\min}$ reveals the resolution of c_i. The case $\Delta \le \Delta_{\min}$ means that there may be a big fluctuation in the cost function. But considering the computation time, we just set the "optimal" value as it was.

Table 1. Pseudocode for Optimizing a Variable c_i

Procedure
Set up: input coefficient vector **c** and coefficient index i
Step 1: set trust-region radius Δ and evaluate
$b_1 = B([\ldots, c_i - \Delta, \ldots])$;
$b_2 = B([\ldots, c_i, \ldots])$;
$b_3 = B([\ldots, c_i + \Delta, \ldots])$.
Step 2: **switch**
case $b_1 \ge b_2 > b_3$: $c_i^* = c_i + \Delta$;
case $b_1 > b_3 \ge b_2$: $c_i^\circ = c_i + (b_1 - b_3)\Delta/2(b_1 - 2b_2 + b_3)$,
if $B([\ldots, c_i^\circ, \ldots]) \le b_2$: $c_i^* = c_i^\circ$;
else if $\Delta \le \Delta_{\min}$: $c_i^* = c_i$, go to **Stop**.
else: $\Delta = \Delta/2$, return to **Step 1**.
case $b_1 = b_3 \ge b_2$: $c_i^* = c_i$;
case $b_2 > b_1 > b_3$: $c_i^* = c_i + \Delta$;
case $b_2 > b_3 \ge b_1$: $c_i^* = c_i - \Delta$;
case $b_3 > b_2 > b_1$: $c_i^* = c_i - \Delta$;
case $b_3 > b_1 \ge b_2$: $c_i^\circ = c_i + (b_1 - b_3)\Delta/2(b_1 - 2b_2 + b_3)$,
if $B([\ldots, c_i^\circ, \ldots]) \le b_2$: $c_i^* = c_i^\circ$;
else if $\Delta \le \Delta_{\min}$: $c_i^* = c_i$, go to **Stop**.
else: $\Delta = \Delta/2$, return to **Step 1**.
Stop: output the optimized coefficient c_i^*.

Table 2. Pseudocode for Optimizing Coefficient Vector c

Procedure
Set up: set initial coefficient vector $\mathbf{c}^0$, trust-region radius Δ, iteration index $k = 0$, and max iteration number K.
While $k < K$:
For c_i^k **in** $\mathbf{c}^k$ **do**
$c_i^k = \arg\min_{x_i \in [c_i^k - \Delta, c_i^k + \Delta]} B([\ldots, x_i, \ldots])$;
$\mathbf{c}^{k+1} = \mathbf{c}^k$,
$k = k + 1$;
Stop: output the optimized coefficient vector $\mathbf{c}^* = \mathbf{c}^k$.

We present the procedure of solving SO, combined with the subproblem, in Table 2.

From the procedure of SO, the cost function evaluation instead of its gradient is required in the proposed method. Also, the proposed method does not depend on the form of cost function, so various metrics can be adopted. The proposed method has a computational complexity of $\mathcal{O}(N)$, where N is the number of source variables. So it is important to devise a source representation method with fewer variables while remaining complete in the solution space.

5. SIMULATIONS

Simulations were carried out with in-house software. The value of CD and NILS and the process latitude are rigorously evaluated by the built-in functions in the lithography simulator, Dr.LiTHO [34]. The wavelength in the simulations was set to 193 nm, the NA was 1.35, and the reduction factor of the projection system is 4. All the position weight μ_p in the CD error metric Eq. (8) and η_p in NILS metric Eq. (9) were set to 1, the factor τ_p in NILS metric Eq. (9) was set to 0.5, and the weight α, β, and γ in the cost function Eq. (11) were, respectively, set to 1, 1, and 500. We set the trust-region radius Δ to 0.1, minimum radius $\Delta_{\min}$ to 0.025, and the maximum iteration number to 10. Since mask topography would cause a best focus shift [12,13], we simply set the out-of-focus value h in Eq. (8) to 0 nm.

We first demonstrate our method for a grating pattern. The mask absorber is composed of a 68 nm molybdenum silicide layer (refractive index = 2.343, extinction coefficient = −0.586), as shown in Fig. 2(a). In this case, the light direction was set to y polarization. With the dipole source [Fig. 2(b)] as the initial guess, we compute the source [Fig. 2(c)] by the proposed method. From Fig. 2(d), it is interesting to find that some of the optimized coefficients, such as c_0, c_6, etc., are dramatically different from the initial one, while some, such as c_2, c_5, etc., remain the same after optimization. It indicates that we can ignore some specific coefficients during the optimization process based on the simulation experience. This will further reduce computing time.

From Fig. 3, the optimized source [Fig. 2(c)] improves imaging performance compared with the dipole source [Fig. 2(b)], specifically, CD from 28.15 to 31.99 nm (desired CD is 32 nm) and NILS from 0.28 to 0.43. The process latitude of these two source patterns is depicted in Fig. 4. It is shown that the optimized source significantly enlarges the process

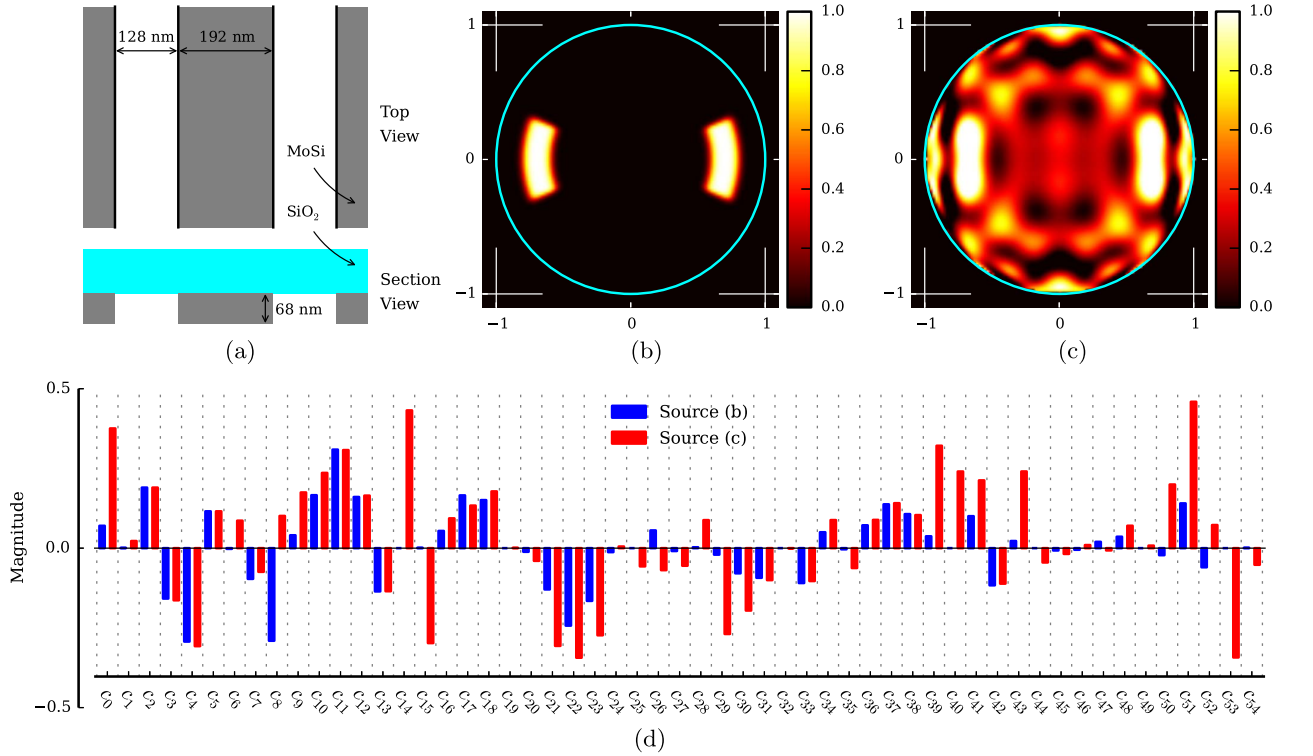

Fig. 2. Simulation results for a grating pattern. (a) Top and section view of 3D mask topography of a grating pattern. It should be noted that its CD is 192 nm/4 = 32 nm since the reduction factor of the projection system is 4. (b) Dipole source (σ_{in}/σ_{out}/openangle = 0.6/0.8/22.5°). (c) Optimized source by the proposed method. (d) Coefficient vectors of source (a) and (b), respectively.

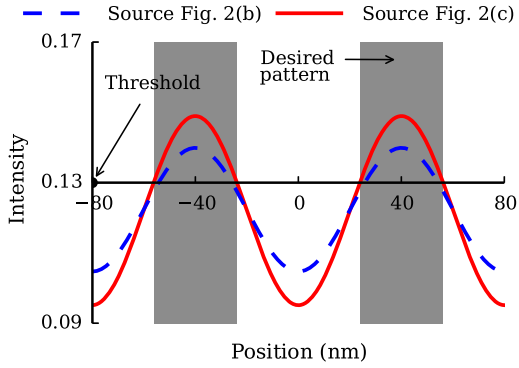

Fig. 3. Intensity comparison of the grating pattern between the dipole source [Fig. 2(b)] and the optimized source [Fig. 2(c)]. The intensity profiles are through the horizontal line spanning two pitches of the mask pattern [Fig. 2(a)] on the wafer side. Resist threshold was set to 0.13. Gray area denotes the desired pattern.

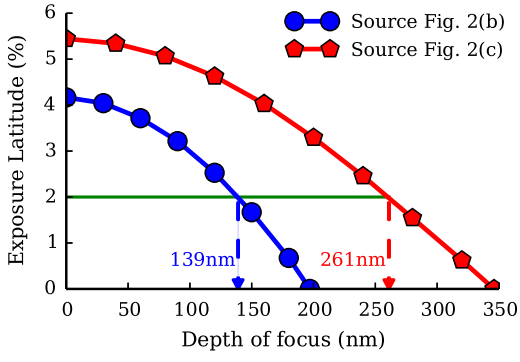

Fig. 4. Process latitude comparison between the dipole source [Fig. 2(b)] and the optimized source [Fig. 2(c)].

latitude. At an exposure latitude of 2%, the depth of focus is expanded from 139 nm to 261 nm, with 87.8% improvement.

Figure 5 shows the convergence history of the proposed method. It is observed that the proposed method converges fast at the first few iterations and becomes slower afterward, which is similar to gradient-based methods [24,25,31]. In this case, it takes 10 iterations; each iteration needs to optimize 55 coefficients, and each coefficient optimization requires at least three cost-function evaluations, so it is estimated that it will take overall $10 \times 55 \times 3 = 1650$ cost function evaluations. The actual number is 3893, which is much more than the estimated value of 1650. The discrepancy between the estimated and actual number of function evaluations is due to the adjustment of the trust-region radius. Reducing the

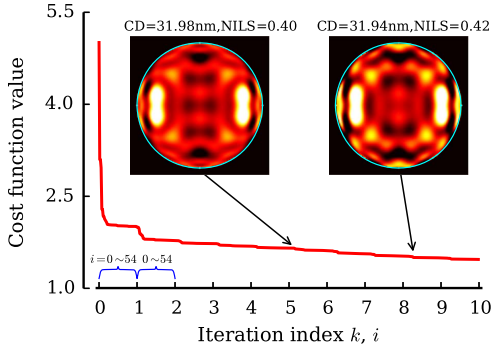

Fig. 5. Convergence history of the proposed method. The major ticks of the horizontal axis denote the iteration number k, and the minor ticks denote the coefficient index i in coefficient vector $\mathbf{c}$, as defined in Table 2. The source patterns shown inside the figure are obtained at iteration index ($k = 5, i = 11$) and ($k = 8, i = 16$), respectively.

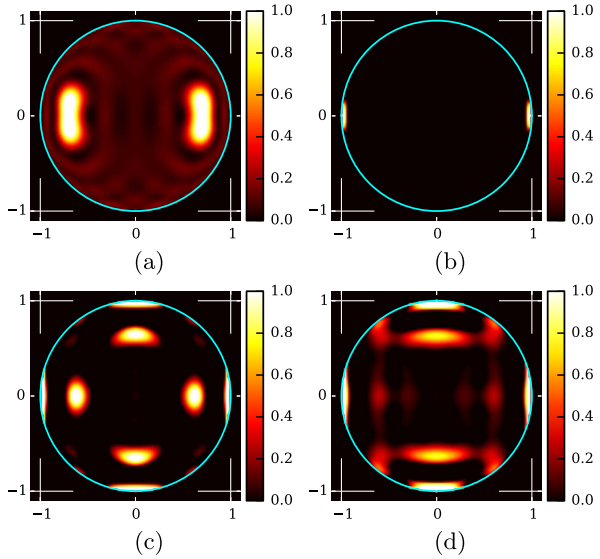

Fig. 6. Optimized sources under the cost function [Eq. (11)] with different weight combinations, where (a) $[\alpha,\beta,\gamma]=[1,0,0]$, (b) $[\alpha,\beta,\gamma]=[0,200,0]$, (c) $[\alpha,\beta,\gamma]=[1,200,0]$, and (d) $[\alpha,\beta,\gamma]=[1,200,500]$, with the same initial pattern [Fig. 2(b)] for the grating pattern [Fig. 2(a)]. The CD under the source patterns (a), (b), (c), and (d), are 32.00 nm, 67.48 nm, 36.03 nm, and 35.94 nm, respectively. The desired CD is 32 nm. The NILS under the source patterns (a), (b), (c), and (d), are 0.35, 1.52, 0.59, and 0.57, respectively.

radius will, in turn, slow down the convergence rate and increase the iteration number and also raise the number of cost function evaluation. One way to avoid this situation is setting a different trust-region radius for a different coefficient. In addition, on the stopping criteria of iteration, an utilitarian strategy is to stop the optimization when some metrics met; for example, CD reaches an accepted range, and NILS is larger than a reasonable value. Two intermediate source patterns, which are obtained after its NILS reaching 0.40 and 0.42, are shown in Fig. 5. For simplicity, we set a limit of the iteration number to stop optimization in this work.

In addition, we investigate the impact of different weight combinations in cost function [Eq. (11)] on a computed source in Fig. 6. The source [Fig. 6(a)] reaches a CD error of 0 after two iterations. It is observed that the dominant feature of Fig. 6(a) is similar to its initial pattern in Fig. 2(b), which reveals that the form of the computed source under CD error metric depends on its initial pattern. The source [Fig. 6(b)] reaches a NILS of 1.52 after four iterations, but it meanwhile results in a CD of 67.48 nm, which is much larger than the desired CD 32 nm. The computed source under NILS metric tends to be a small dipole pattern for the grating mask pattern. Comparing Fig. 6(d) with Fig. 6(c), the proposed source quality metric shows capacity of forming a smooth source pattern. From these simulations, it is demonstrated that the form of source depends heavily on the cost function definition and the initial pattern. In mathematics, when the cost function of a problem is defined and an initial guess is given, the local minima near the initial guess are definite. The proposed method is a solution method aimed at finding one of these minima.

Another set of simulations for a brick array is shown in Fig. 7. We employ a chrome-on-glass mask with an absorber

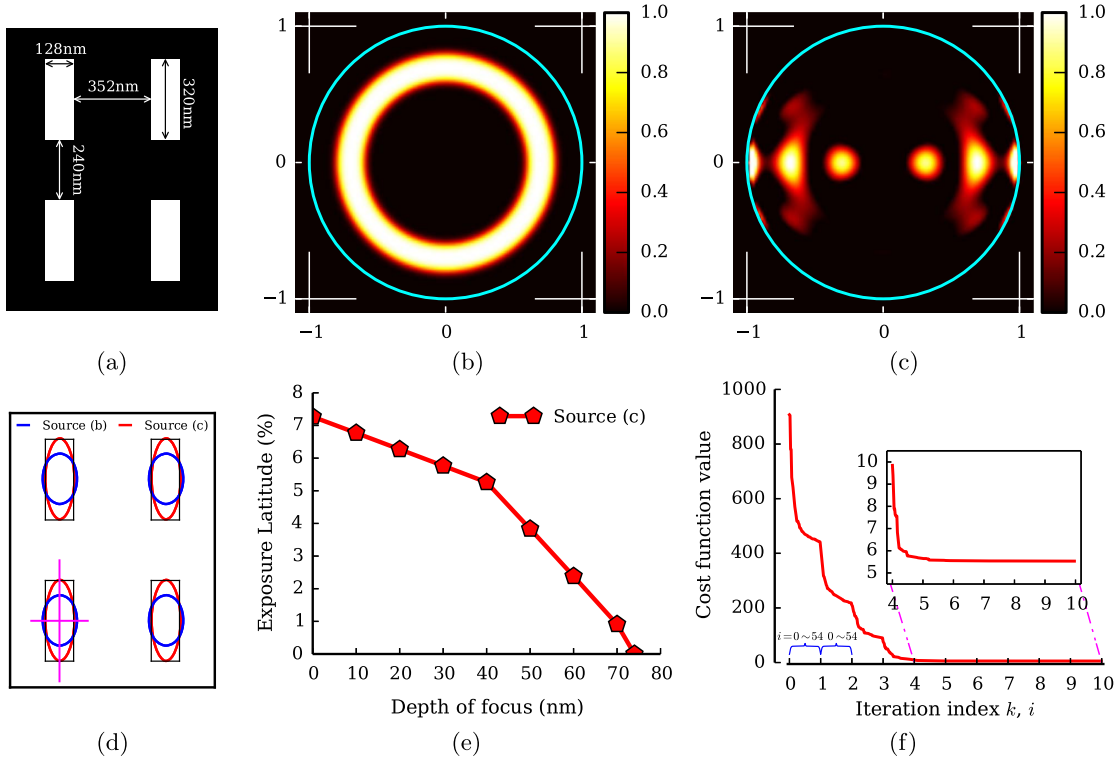

Fig. 7. Simulation results for a brick array pattern. (a) Regular brick array. (b) Annular source ($\sigma_{in}/\sigma_{out}=0.6/0.8$). (c) Optimized source by the proposed method. (d) Contour comparison between the annular source (b) and the optimized source (c) on the wafer side. The black line denotes the desired contour. The magenta lines across the brick are CD cutlines. (e) Process latitude of the optimized source (c). Process latitude is unavailable with the annular source (b). (f) Convergence history of the proposed method. The major ticks of the horizontal axis denote the iteration number k, and the minor ticks denote the coefficient index i in coefficient vector $\mathbf{c}$, as defined in Table 2. The inset is a zoomed view of the convergence history of iteration index k from 4 to 10.

layer of 80 nm (refractive index = 0.861, extinction coefficient = −1.688). In this case, the light direction was set to xy polarization, and we took two cutlines across the brick as the magenta lines shown in Fig. 7(d). With the annular source [Fig. 7(b)] as the initial guess, it computes the source [Fig. 7(c)] by the proposed method. From Fig. 7(d), the optimized source results in an image contour closer to the desired one. The CD along the horizontal and vertical cutlines by the optimized source pattern is, respectively, 32.33 nm and 79.91 nm (desired CD is 32 nm and 80 nm), compared with 38.44 nm and 49.77 nm by the annular source [Fig. 7(b)]. The process latitude under the annular source [Fig. 7(b)] is unavailable, while it becomes available under the optimized source [Fig. 7(c)]. Overall, it takes 10 iterations and 3776 cost function evaluations to obtain the source.

These simulations demonstrate that the proposed method can handle SO under an accurate model and has the capacity to optimize a source pattern with considerable improvement in lithography performance in terms of CD error, NILS, and process latitude. As shown in Figs. 5 and 7(f), it is observed that the cost function descends rhythmically; that is, the descent rate of the cost function with respect to the first few coefficients is faster than the late coefficients, and this tendency is maintained in each phase [iteration k in Figs. 5 and 7(f)]. This also indicates that the coefficients of the first few basis functions contribute a lot to the descent of the cost function. The reason would be that the first few kernels contain the primary pattern, while the late kernels contain a small and detailed pattern, as depicted in Fig. 1. So it is reasonable to optimize the first few coefficients in SO, instead of considering all the coefficients.

6. CONCLUSIONS

In conclusion, we propose an improved kernel-based source pattern representation method that has moderate parameters and considers the non-negative and bounded property of practical source intensity. As a result, we do not have to consider these regularities during the SO process. Then, we formulate the SO problem as a minimization problem by means of various metrics. Finally, we develop a DFO method to solve the SO problem. This method is independent of the form of the cost function and has the capacity of optimizing a source with considerable improvement in lithography performance in terms of CD error, NILS, and process latitude.

The paper also provides a strategy to deal with general ill-posed problems in optical lithography such as source optimization, mask optimization, calibration, and optical proximity effect matching. The strategy is that we dice the problem into subproblems in a relatively small region in which the subproblem is well-posed and can be more easily handled, instead of tackling the original one directly.

ACKNOWLEDGMENTS

This work was funded by the National Natural Science Foundation of China (grant nos. 51475191 and 51405172), the Specialized Research Fund for the Doctoral Program of Higher Education of China (grant no. 20120142110019), the National Science and Technology Major Project of China (grant no. 2012ZX02701001), and the Program for Changjiang Scholars and Innovative Research Team in University of China. It was also supported in part by the UGC Areas of Excellence project Theory, Modeling, and Simulation of Emerging Electronics, and by the State Key Lab of Digital Manufacturing Equipment and Technology under project DMETKF2013003. The authors would also like to acknowledge the Fraunhofer Institute for Integrated Systems and Device Technology for providing the use of Dr.LiTHO software.

REFERENCES

1. A. K. Wong, *Resolution Enhancement Technologies in Optical Lithography* (SPIE, 2001).
2. E. Y. Lam and A. K. Wong, "Computation lithography: virtual reality and virtual virtuality," Opt. Express **17**, 12259–12268 (2009).
3. D. O. S. Melville, A. E. Rosenbluth, A. Waechter, M. Millstone, J. Tirapu-Azpiroz, K. Tian, K. Lai, T. Inoue, M. Sakamoto, K. Adam, and A. Tritchkov, "Computational lithography: exhausting the resolution limits of 193-nm projection lithography systems," J. Vac. Sci. Technol. B **29**, 06FH04 (2011).
4. M. Burkhardt, A. Yen, C. Progler, and G. Wells, "Illuminator design for the printing of regular contact patterns," Microelectron. Eng. **41–42**, 91–95 (1998).
5. R. Socha, M. Eurlings, F. Nowak, and J. Finders, "Illumination optimization of periodic patterns for maximum process window," Microelectron. Eng. **61–62**, 57–64 (2002).
6. J.-C. Yu, P. Yu, and H.-Y. Chao, "Library-based illumination synthesis for critical CMOS patterning," IEEE Trans. Image Process. **22**, 2811–2821 (2013).
7. A. E. Rosenbluth, S. Bukofsky, C. Fonseca, M. Hibbs, K. Lai, A. F. Molless, R. N. Singh, and A. K. Wong, "Optimum mask and source patterns to print a given shape," J. Microlith. Microfab. Microsyst. **1**, 13–30 (2002).
8. X. Ma and G. R. Arce, "Pixel-based simultaneous source and mask optimization for resolution enhancement in optical lithography," Opt. Express **17**, 5783–5793 (2009).
9. J. Li, S. Liu, and E. Y. Lam, "Efficient source and mask optimization with augmented Lagrangian methods in optical lithography," Opt. Express **21**, 8076–8090 (2013).
10. J. Li and E. Y. Lam, "Robust source and mask optimization compensating for mask topography effects in computational lithography," Opt. Express **22**, 9471–9485 (2014).
11. Y. Granik, "Source optimization for image fidelity and throughput," J. Microlith. Microfab. Microsyst. **3**, 509–522 (2004).
12. M. K. Sears, G. Fenger, J. Mailfert, and B. Smith, "Extending SMO into the lens pupil domain," Proc. SPIE **7973**, 79731B (2011).
13. T. Fühner, P. Evanschitzky, and A. Erdmann, "Mutual source, mask and projector pupil optimization," Proc. SPIE **8326**, 83260I (2012).
14. T. D. Milster, H. Noble, E. Ford, W. Dallas, R. A. Chipman, I. Matsubara, Y. Unno, S. McClain, P. Khulbe, W. S. T. Lam, and D. Hansen, "Polarization holograms for source-mask optimization," Proc. SPIE **7973**, 79731A (2011).
15. K. Tian, A. Krasnoperova, D. Melville, A. E. Rosenbluth, D. Gil, J. Tirapu-Azpiroz, K. Lai, S. Bagheri, C.-C. Chen, and B. Morgenfeld, "Benefits and trade-offs of global source optimization in optical lithography," Proc. SPIE **7274**, 72740C (2009).
16. K. Lai, A. E. Rosenbluth, S. Bagheri, J. Hoffnagle, K. Tian, D. Melville, J. Tirapu-Azpiroz, M. Fakhry, Y. Kim, S. Halle, G. McIntyre, A. Wagner, G. Burr, M. Burkhardt, D. Corliss, E. Gallagher, T. Faure, M. Hibbs, D. Flagello, J. Zimmermann, B. Kneer, F. Rohmund, F. Hartung, C. Hennerkes, M. Maul, R. Kazinczi, A. Engelen, R. Carpaij, R. Groenendijk, J. Hageman, and C. Russ, "Experimental result and simulation analysis for the use of pixelated illumination from source mask optimization for 22-nm logic lithography process," Proc. SPIE **7274**, 72740A (2009).
17. J.-C. Yu, P. Yu, and H.-Y. Chao, "Fast source optimization involving quadratic line-contour objectives for the resist image," Opt. Express **20**, 8161–8174 (2012).
18. N. Jia and E. Y. Lam, "Pixelated source mask optimization for process robustness in optical lithography," Opt. Express **19**, 19384–19398 (2011).

19. T. Fühner, A. Erdmann, and S. Seifert, "Direct optimization approach for lithographic process conditions," J. Micro/Nanolith. MEMS MOEMS **6**, 031006 (2007).
20. L. Pang, P. Hu, D. Peng, D. Chen, T. Cecil, L. He, G. Xiao, V. Tolani, T. Dam, K.-H. Baik, and B. Gleason, "Source mask optimization (SMO) at full chip scale using inverse lithography technology (ILT) based on level set methods," Proc. SPIE **7520**, 75200X (2009).
21. X. Wu, S. Liu, J. Li, and E. Y. Lam, "Efficient source mask optimization with zernike polynomial functions for source representation," Opt. Express **22**, 3924–3937 (2014).
22. Z. Song, X. Ma, J. Gao, J. Wang, Y. Li, and G. R. Arce, "Inverse lithography source optimization via compressive sensing," Opt. Express **22**, 14180–14198 (2014).
23. A. K. Wong, *Optical Imaging in Projection Microlithography* (SPIE, 2005).
24. W. Lv, Q. Xia, and S. Liu, "Mask-filtering-based inverse lithography," J. Micro/Nanolith. MEMS MOEMS **12**, 043003 (2013).
25. W. Lv, S. Liu, Q. Xia, X. Wu, Y. Shen, and E. Y. Lam, "Level-set-based inverse lithography for mask synthesis using the conjugate gradient and an optimal time step," J. Vac. Sci. Technol. B **31**, 041605 (2013).
26. A. Erdmann, T. Fühner, F. Shao, and P. Evanschitzky, "Lithography simulation: modeling techniques and selected applications," Proc. SPIE **7390**, 739002 (2009).
27. X. Ma, C. Han, Y. Li, L. Dong, and G. R. Arce, "Pixelated source and mask optimization for immersion lithography," J. Opt. Soc. Am. A **30**, 112–123 (2013).
28. A. Erdmann, T. Fühner, T. Schnattinger, and B. Tollkühn, "Toward automatic mask and source optimization for optical lithography," Proc. SPIE **5377**, 646–657 (2004).
29. A. R. Conn, K. Scheinberg, and P. L. Toint, "Global convergence of general derivative-free trust-region algorithms to first-and second-order critical points," SIAM J. Opt. **20**, 387–415 (2009).
30. A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization* (SIAM, 2009).
31. W. Lv, E. Y. Lam, H. Wei, and S. Liu, "Cascadic multigrid algorithm for robust inverse mask synthesis in optical lithography," J. Micro/Nanolith. MEMS MOEMS **13**, 023003 (2014).
32. P. Gong, S. Liu, W. Lv, and X. Zhou, "Fast aerial image simulations for partially coherent systems by transmission cross coefficient decomposition with analytical kernels," J. Vac. Sci. Technol. B **30**, 06FH03 (2012).
33. E. Barouch, S. L. Knodle, S. A. Orszag, and M. S. Yeung, "Illuminator optimization for projection printing," Proc. SPIE **3679**, 697–703 (1999).
34. "Dr.LiTHO," http://www.drlitho.com/ (June 2014).