Efficient source and mask optimization with augmented Lagrangian methods in optical lithography

Jia Li,[1] **Shiyuan Liu,**[2] **and Edmund Y. Lam**[1,*]

[1]*Imaging Systems Laboratory, Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong, China*
[2]*State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, China*
[*]*elam@eee.hku.hk*

Abstract: Source mask optimization (SMO) is a powerful and effective technique to obtain sufficient process stability in optical lithography, particularly in view of the challenges associated with 22 nm process technology and beyond. However, SMO algorithms generally involve computation-intensive nonlinear optimization. In this work, a fast algorithm based on augmented Lagrangian methods (ALMs) is developed for solving SMO. We first convert the optimization to an equivalent problem with constraints using variable splitting, and then apply an alternating minimization method which gives a straightforward implementation of the algorithm. We also use the quasi-Newton method to tackle the sub-problem so as to accelerate convergence, and a tentative penalty parameter schedule for adjustment and control. Simulation results demonstrate that the proposed method leads to faster convergence and better pattern fidelity.

OCIS codes: (110.3960) Microlithography; (110.5220) Photolithography; (110.1758) Computational imaging.

References and links

1. B. Küchler, A. Shamsuarov, T. Mülders, U. Klostermann, S.-H. Yang, S. Moon, V. Domnenko, and S.-W. Park, "Computational process optimization of array edges," in *Optical Microlithography XXV*, W. Conley, ed. (2012), vol. 8326 of *Proc. SPIE*, p. 83260H.
2. 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).
3. M. Fakhry, Y. Granik, K. Adam, and K. Lai, "Total source mask optimization: high-capacity, resist modeling, and production-ready mask solution," in *Photomask Technology 2011*, W. Maurer and F. E. Abboud, eds. (2011), vol. 8166 of *Proc. SPIE*, p. 81663M.
4. N. Jia and E. Y. Lam, "Pixelated source mask optimization for process robustness in optical lithography," Opt. Express **19**, 19384–19398 (2011).
5. S. K. Choy, N. Jia, C. S. Tong, M. L. Tang, and E. Y. Lam, "A robust computational algorithm for inverse photomask synthesis in optical projection lithography," SIAM J. Imaging Sciences **5**, 625–651 (2012).
6. Y. Peng, J. Zhang, Y. Wang, and Z. Yu, "Gradient-based source and mask optimization in optical lithography," IEEE Trans. Image Process. **20**, 2856–2864 (2011).
7. Y. Deng, Y. Zou, K. Yoshimoto, Y. Ma, C. E. Tabery, J. Kye, L. Capodieci, and H. J. Levinson, "Considerations in source-mask optimization for logic applications," in *Optical Microlithography XXIII*, M. V. Dusa and W. Conley, eds. (2010), vol. 7640 of *Proc. SPIE*, p. 7640J.
8. D. Zhang, G. Chua, Y. Foong, Y. Zou, S. Hsu, S. Baron, M. Feng, H.-Y. Liu, Z. Li, S. Jessy, T. Yun, C. Babcock, C. B. IL, R. Stefan, A. Navarra, T. Fischer, A. Leschok, X. Liu, W. Shi, J. Qiu, and R. Dover, "Source mask optimization methodology (SMO) and application to real full chip optical proximity correction," in *Optical Microlithography XXV*, W. Conley, ed. (2012), vol. 8326 of *Proc. SPIE*, p. 83261V.

#184602 - $15.00 USD
Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

9. K. Iwase, P. D. Bisschop, B. Laenens, Z. Li, K. Gronlund, P. V. Adrichem, and S. Hsu, "A new source optimization approach for 2X node logic," in *Photomask Technology 2011*, W. Maurer and F. E. Abboud, eds. (2011), vol. 8166 of *Proc. SPIE*, p. 81662A.
10. 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).
11. J. Li, Y. Shen, and E. Y. Lam, "Hotspot-aware fast source and mask optimization," Opt. Express **20**, 21792–21804 (2012).
12. Y. Shen, N. Wong, and E. Y. Lam, "Level-set-based inverse lithography for photomask synthesis," Opt. Express **17**, 23690–23701 (2009).
13. L. Pang, G. Xiao, V. Tolani, P. Hu, T. Cecil, T. Dam, K.-H. Baik, and B. Gleason, "Considering MEEF in inverse lithography technology (ILT) and source mask optimization (SMO)," in *Photomask Technology*, H. Kawahira and L. S. Zurbrick, eds. (2008), vol. 7122 of *Proc. SPIE*, p. 71221W.
14. Y. Shen, N. Jia, N. Wong, and E. Y. Lam, "Robust level-set-based inverse lithography," Opt. Express **19**, 5511–5521 (2011).
15. J.-C. Yu and P. Yu, "Gradient-based fast source mask optimization (SMO)," in *Optical Microlithography XXIV*, M. V. Dusa, ed. (2011), vol. 7973 of *Proc. SPIE*, p. 797320.
16. N. Jia and E. Y. Lam, "Machine learning for inverse lithography: using stochastic gradient descent for robust photomask synthesis," J. Opt. **12**, 045601 (2010).
17. S. H. Chan, A. K. Wong, and E. Y. Lam, "Initialization for robust inverse synthesis of phase-shifting masks in optical projection lithography," Opt. Express **16**, 14746–14760 (2008).
18. S. H. Chan and E. Y. Lam, "Inverse image problem of designing phase shifting masks in optical lithography," in *IEEE International Conference on Image Processing*, (2008), p. 1832–1835.
19. E. Y. Lam and A. K. Wong, "Computation lithography: Virtual reality and virtual virtuality," Opt. Express **17**, 12259–12268 (2009).
20. E. Y. Lam and A. K. Wong, "Nebulous hotspot and algorithm variability in computation lithography," J. Micro/Nanolith., MEMS, MOEMS **9**, 033002 (2010).
21. J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. (Springer, 2006).
22. M. R. Hestenes, "Multiplier and gradient methods," J. Optimiz. Theory App. **4**, 303–320 (1969).
23. M. Powell, "A method for nonlinear constraints in minimization problems," in *Optimization*, R. Fletcher, ed. (1969), *Academic*, p. 283–298.
24. M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems," IEEE Trans. Image Process. **20**, 681–695 (2011).
25. S. Ramani and J. A. Fessler, "Parallel MR image reconstruction using augmented Lagrangian methods," IEEE Trans. Image Process. **30**, 694–706 (2011).
26. S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," Found. Trends Mach. Learn. **3**, 1–124 (2011).
27. R. T. Rockafellar, "Augmented Lagrange multiplier functions and duality in nonconvex programming," SIAM J. Control **12**, 268–285 (1974).
28. N. B. Cobb, "Fast optical and process proximity correction algorithms for integrated circuit manufacturing," Ph.D. thesis, Univ. of California at Berkeley, Berkeley, California (1998).
29. A. K. Wong, *Optical Imaging in Projection Microlithography* (SPIE, 2005).
30. A. Poonawala and P. Milanfar, "Mask design for optical microlithography— an inverse imaging problem," IEEE Trans. Image Process. **16**, 774–788 (2007).
31. T. Goldstein and S. Osher, "The split Bregman algorithm for l_1 regularized problems," SIAM J. Imaging Sciences **2**, 323–343 (2009).
32. M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," IEEE Trans. Image Process. **19**, 2345–2356 (2010).
33. J. L. Morales and J. Nocedal, "Remark on 'algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization'," ACM Trans. Math Software **23**, 550–560 (2011).
34. S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, "An augmented Lagrangian method for total variation video restoration," IEEE Trans. Image Process. **20**, 14746–14760 (2011).
35. D. Noll, "Local convergence of an augmented Lagrangian method for matrix inequality constrained programming," Optim. Method Softw. **22**, 777–802 (2007).
36. D. K. Bertsekas, *Constrained Optimization and Lagrange Multiplier Method* (Academic, 1982).

1. Introduction

The limitation in resolution capacity of physical lithography tools and the ever growing integration density of semiconductor devices lead to reduced pattern fidelity and smaller process window (PW) in optical lithography [1], particularly for the 22 nm feature generation and beyond.

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

As an integral part of advanced computational lithography techniques, source mask optimization (SMO) has been considered a promising candidate to tackle these challenges and enable the continuation of current immersion lithography [2, 3].

The main goal of the SMO approach is to achieve a pair of optimal source and mask, which together ensure that a higher image fidelity and improved performance on aberration stability are obtained. The cost function of SMO typically measures the image fidelity in terms of edge placement errors (EPEs) or the deviation of the printed image from the desired one [4, 5], and evaluates the process robustness by depth of focus (DoF) [6], process window or other factors as deemed appropriate [7]. This objective function can also strike a moderate balance between the conflicting optimization performances, such as the small mask error enhancement factor (MEEF) and large DoF [8]. Moreover, source optimization during SMO provides more flexibility regarding both the source profile and its intensity [9, 10], improving the process margin on the wafer. Besides these desirable features, the SMO process can also focus on the critical patterns and obtain a sufficiently large process window for these hotspot regions [11]. Most existing SMO algorithms pose the mask synthesis and source design as an inverse problem solved by iterative methods [12], including level-set method [13, 14] and different gradient-based approaches, like steepest descent algorithm [15, 16] and conjugate gradient method [4, 17, 18].

Unfortunately, although these methods have been applied to tackle the constrained optimization problem within the inverse imaging calculations, they are normally computationally intensive, resulting in slow convergence and limiting the wide adoption of SMO for practical full-chip circuits patterns simulations [19, 20]. Additionally, the computed free-form source and pixelated mask design can be too complex to fulfill manufacturing constraints. To address these two issues, in this paper we propose an efficient SMO algorithm based on augmented Lagrangian methods (ALMs). ALMs are a certain class of algorithms for solving constrained optimization problems, which replace the original optimization problem with a series of unconstrained problems, reducing the possibility of a widely changing objective function by introducing the Lagrange multiplier into the augmented Lagrangian function [21]. The study of ALMs dates back to as early as the late 1960s [22, 23], yet recent developments that incorporate sparse matrix techniques and the use of partial updates have rekindled a lot of interest in this approach [24–26]. This is especially true in solving constrained optimization problems, due to its flexible problem formulation, desirable convergence property and avoidance of the ill-conditional behavior, and its global convergence for non-convex optimization problems [27]. ALMs can be implemented by general-purpose software packages (e.g., LANCELOT and MINOS) or tailored for specific purposes.

This paper focuses on a fast SMO algorithm using inverse synthesis based on ALMs, and the major contributions are threefold. First, in terms of image quality, we demonstrate a better algorithmic performance with fewer pattern errors, better normalized image log slopes (NILS) and larger process window sizes, based on simulation results with gate and poly patterns. Second, in terms of processing time, we achieve a higher speed compared to other commonly-used methods. This results from the use of the quasi-Newton method and an updated scheme of the penalty parameter, which are iteratively computed to find solutions to two sub-problems. Our method can improve the convergence rate, thereby shortening the overall execution time. Third, in terms of manufacturability, our algorithm is able to generate low-complexity source and mask patterns. This is fulfilled by constructing an augmented Lagrangian function, where we incorporate the complexity penalty as an equality constraint, resulting in a bound-constrained nonlinear optimization problem that can be solved by minimizing alternatively with respect to one auxiliary variable at a time.

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

2. Forward imaging model

In optical lithography, one has in mind a particular desired pattern to print on the wafer, and every effort is then made to set up the system properly such that the actual printed image is very close. Therefore, a critical step in computational lithography is to model this imaging process accurately, with the various parameters available for adjustment. Often, we discretize the images to a size $N \times N$; furthermore, for ease of description, we turn them into vectors of length N^2 through lexicographic ordering. In what follows, we use $\mathbf{z}$ for the actual printed image and $\mathbf{z}_0$ for the desired pattern, respectively. In addition, we use $\mathbf{m}$ to denote the mask that produces the circuit pattern, which is also a vector of length N^2.

Light intensity is then calculated by the sum of coherent systems (SOCS) model [28], which has shown to be very efficient in mask optimization. Because it takes advantage of Singular Value Decomposition (SVD) to decompose the illumination system into different kernels, the rapidly descending singular values enable the light intensity computation with only the sum of a small number of coherent systems. Let P be the total number of kernels used in the computation, $\tilde{\mathbf{H}}_l$ be the lth kernel, and λ_l be the corresponding singular value. We use the discrete form for image computation, so the vector notation of the aerial image $\mathbf{z}_a$ can be approximated by

$$\mathbf{z}_a \approx \sum_{l=1}^{P} \lambda_l \left\| \tilde{\mathbf{H}}_l * \mathbf{m} \right\|^2, \tag{1}$$

where $*$ denotes convolution.

On the other hand, the gradient to the source cannot be calculated directly using the SOCS model. Instead, we simulate the aerial image using Abbe's method [29] in the source optimization flow, which integrates the images formed by all the source points incoherently. If we use $\mathbf{I}_s$ to represent the image formed by a unit source pixel, $\mathbf{z}'_a$ can be interpreted as a linear superposition of images with source $\mathbf{s}$ as coefficients [10], i.e.,

$$\mathbf{z}'_a = \mathbf{I}_s \, \mathbf{s}. \tag{2}$$

The aerial image then undergoes the photoresist development to form the printed image $\mathbf{z}$. Approximating the resist effect with a sigmoid function due to its differentiability [30], we can derive the output of the lithography system as

$$\mathbf{z} = \frac{1}{1 + \mathrm{e}^{-\alpha(\mathbf{z}_a - t_r)}}, \tag{3}$$

where t_r is the threshold and α indicates the steepness of the sigmoid function.

3. Augmented Lagrangian method for inverse lithography

In this section, we describe how to apply the augmented Lagrangian method to SMO. The optimization procedure consists of the mask and source updates, which are performed alternately until the termination criterion is reached. The target pattern is assigned as the initial mask, and the first mask optimization is performed with a fixed traditional annular illumination. The resulting optimal mask is then used during the following source updates, which output the optimal source to the succeeding mask updates.

For the purpose of discussing the algorithms below, we define the following operators. The first one, $\mathscr{V}(\cdot)$, denotes the vectorization of a matrix, which converts it into a column vector using a lexicographical order. For example, if $\mathbf{M}$ is the mask represented in a matrix form, we can write $\mathbf{m} = \mathscr{V}(\mathbf{M})$ for the mask as a vector. Next are $\mathscr{D}_x$ and $\mathscr{D}_y$, both represent the first-order forward finite difference operators, defined respectively as

$$\mathscr{D}_x(\mathbf{m}) = \mathscr{V}(\mathbf{M}_x - \mathbf{M}) \qquad \text{and} \qquad \mathscr{D}_y(\mathbf{m}) = \mathscr{V}(\mathbf{M}_y - \mathbf{M}), \tag{4}$$

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

where $\mathbf{M}_x$ and $\mathbf{M}_y$ means shifting $\mathbf{M}$ along the horizontal and vertical directions by one pixel, respectively. To write the equations as compact as possible, we also define

$$\mathscr{D} = \begin{bmatrix} \mathscr{D}_x \\ \mathscr{D}_y \end{bmatrix}. \tag{5}$$

3.1. *Mask optimization problem*

To achieve the smallest accumulated pattern error (PE), we generate the optimal mask by minimizing the sum of the mismatches between the printed image and the desired one over all locations, together with a regularization term to reduce the mask complexity. We constrain the local variation of the difference between the mask and the desired pattern [30], suppressing the small-scale protrusion while preserving the large-scale features. Mathematically, the cost function of the mask optimization problem is formulated as

$$\frac{\mu}{2}\left\|\mathbf{z}-\mathbf{z}_0\right\|_2^2 + \left\|\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right\|_1. \tag{6}$$

Here, μ is a parameter for the tradeoff between the pattern fidelity term and the regularization term.

In order to solve Eq. (6) by the augmented Lagrangian scheme, we first transform it to an equivalent constrained optimization problem. An intermediate variable $\mathbf{v}$ is created by variable splitting, which has been recently used in several image processing applications [31, 32]. The rationale behind the variable splitting method is that it can be easier to solve the constrained problem than to solve its original unconstrained counterpart [24]. In addition, unlike other imaging processing problems, we need the solution to correspond to a binary mask pattern. To do so, we relax the mask pixel values to take on values between 0 and 1 inclusive, which is added as a constraint to the optimization. This leads to the following constrained problem

$$\begin{aligned} \underset{\mathbf{m}}{\text{minimize}} \quad & f_1(\mathbf{m}) = \frac{\mu}{2}\left\|\mathbf{z}-\mathbf{z}_0\right\|_2^2 + \left\|\mathbf{v}\right\|_1 \\ \text{subject to} \quad & \mathbf{v} = \mathscr{D}(\mathbf{m}-\mathbf{z}_0), \\ & 0 \le \mathbf{m} \le 1. \end{aligned} \tag{7}$$

To solve this optimization problem, we first derive the augmented Lagrangian function,

$$L_\rho(\mathbf{m},\mathbf{v},\mathbf{d}) = \frac{\mu}{2}\left\|\mathbf{z}-\mathbf{z}_0\right\|_2^2 + \left\|\mathbf{v}\right\|_1 - \mathbf{d}^T\left[\mathbf{v}-\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right] + \frac{\rho}{2}\left\|\mathbf{v}-\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right\|_2^2. \tag{8}$$

Here, the Lagrange multiplier $\mathbf{d}$ associated with the constraint $\mathbf{v} = \mathscr{D}(\mathbf{m}-\mathbf{z}_0)$ and penalty parameter ρ are introduced. The Lagrange multiplier is used to find the extrema of a multivariable function $f(x,y)$ subject to a constraint $g(x) = c$, by locating where the gradient of f is parallel to the gradient of g. Note that ALM works by alternating between minimizing the primal (**m**) given its dual (**d**), and maximizing the dual by keeping its primal function fixed [21], and repeating these two steps until a stopping criterion is satisfied. In our context, the primal minimization step is fulfilled by updating two variables because of the auxiliary variable we introduced, so an algorithm known as the alternating direction method (ADM) is employed to solve the following sub-problems iteratively:

$$\begin{aligned} \mathbf{m}_{k+1} = \arg\min_{\mathbf{m}} \quad & \frac{\mu}{2}\left\|\mathbf{z}-\mathbf{z}_0\right\|_2^2 - \mathbf{d}_k^T\left[\mathbf{v}_k-\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right] + \frac{\rho}{2}\left\|\mathbf{v}_k-\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right\|_2^2 \\ \text{subject to} \quad & 0 \le \mathbf{m} \le 1, \end{aligned} \tag{9}$$

$$\mathbf{v}_{k+1} = \arg\min_{\mathbf{v}} \quad \left\|\mathbf{v}\right\|_1 - \mathbf{d}_k^T\left[\mathbf{v}-\mathscr{D}(\mathbf{m}_{k+1}-\mathbf{z}_0)\right] + \frac{\rho}{2}\left\|\mathbf{v}-\mathscr{D}(\mathbf{m}_{k+1}-\mathbf{z}_0)\right\|_2^2, \tag{10}$$

$$\mathbf{d}_{k+1} = \mathbf{d}_k - \rho\left[\mathbf{v}_{k+1}-\mathscr{D}(\mathbf{m}_{k+1}-\mathbf{z}_0)\right], \tag{11}$$

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

in which the subscript k denotes the kth iteration.

We now investigate these sub-problems in the following subsections.

1. **m**-subproblem: To find the solution to Eq. (9), we need to calculate the gradient of $L_\rho(\mathbf{m},\mathbf{v},\mathbf{d})$ with respect to **m**. Due to the discrete nature of the mask, we define a differential operator $\partial f/\partial \mathbf{m}$ to evaluate the gradient of a function f with respect to its argument **m**, which is approximated by numerical differences. As shown in the Appendix, it is given by

$$\frac{\partial L_\rho(\mathbf{m},\mathbf{v},\mathbf{d})}{\partial \mathbf{m}} = \mu\alpha \mathrm{Re}\left\{ \sum_{l=1}^{P} \lambda_l \left(\bar{\mathbf{H}}_l * \left[(\mathbf{z}-\mathbf{z}_0)\odot \mathbf{z} \odot (1-\mathbf{z}) \odot (\tilde{\mathbf{H}}_l * \mathbf{m})^\dagger \right] \right) \right\} + \mathscr{D}^T(\mathbf{d}_k) - \rho \mathscr{D}^T(\mathbf{v}_k) + \rho \mathscr{D}^T\mathscr{D}(\mathbf{m}) - \rho \mathscr{D}^T\mathscr{D}(\mathbf{z}_0), \tag{12}$$

   where $\bar{\mathbf{H}}_l(x,y) = \tilde{\mathbf{H}}_l(-x,-y)$, and $\odot$ indicates pixel-by-pixel multiplication while symbol $\dagger$ is a complex conjugate operator.

   As can be observed from Eq. (12), the minimization problem of Eq. (9) is not trivial since this gradient involves quartic, non-smooth terms and bound constraints. Hence, one cannot obtain an analytical formula for the minimization step involving **m**, but needs to solve it iteratively. We choose to use an optimization technique called the L-BFGS-B algorithm, which stands for the Limited-memory Broyden-Fletcher-Goldfarb-Shanno method with simple Bounds on the variables [33]. This method is particularly suitable for optimization problems with a large number of variables, due to its moderate memory requirement and independence of the properties of the cost function. More importantly, as a quasi-Newton optimization method, its superior convergence property makes our SMO method promising in large-scale practical applications of inverse lithography.

2. **v**-subproblem: Eq. (10) can be solved using the shrinkage formula [34], and we therefore have

$$\mathbf{v}_{k+1} = \max\left\{ \left| \frac{\mathbf{d}_k}{\rho} + \mathscr{D}(\mathbf{m}_{k+1}-\mathbf{z}_0) \right| - \frac{1}{\rho}, 0 \right\} \odot \mathrm{sgn}\left[\frac{\mathbf{d}_k}{\rho} + \mathscr{D}(\mathbf{m}_{k+1}-\mathbf{z}_0) \right], \tag{13}$$

   where

$$\mathrm{sgn}(x) = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0. \end{cases} \tag{14}$$

3. **d**-subproblem: Multiplier **d** is updated as described in Eq. (11). This is also another merit of ALMs, namely that the optimal step size to update $\mathbf{d}_k$ is determined by the chosen penalty parameter ρ. This enables much easier parameter tuning than the common iterative thresholding methods. On the other hand, unlike the penalty approach, it is not necessary to enforce that ρ approaches infinity to guarantee convergence for the original optimization problem. Instead, the existence of the Lagrange multiplier enables the penalty parameter to take a relatively smaller value, thereby improving the convergence [21]. The method also can be extended to tackle practical inequality constraints problem [35]. Accordingly, choosing an appropriate ρ is a critical issue in ALMs applications, and we will discuss it later in this paper.

The pseudo-code in Table 1 elaborates on the procedure of this proposed algorithm for mask optimization.

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

Table 1. Pseudo-code of mask optimization procedure

Algorithm 1: ALM for mask optimization problem

Input: Initial mask $\mathbf{m}_1 = \mathbf{z}_0$, initial multiplier $\mathbf{d}_1 = 0$;
Choose convergence parameter ε; Set $\rho > 0$, $\tau > 1$, $0 < \alpha < 1$;

for k=1,2,... **do**

1. Solve the subproblem of $\mathbf{m}_{k+1}$ (9) using L-BFGS-B;
2. $\mathbf{v}_{k+1} = \max\left\{\left|\mathbf{d}_k/\rho + \mathscr{D}(\mathbf{m}_{k+1} - \mathbf{z}_0)\right| - 1/\rho, 0\right\} \odot \operatorname{sgn}\left[\mathbf{d}_k/\rho + \mathscr{D}(\mathbf{m}_{k+1} - \mathbf{z}_0)\right]$;
3. Update the Lagrangian multiplier $\mathbf{d}_{k+1} = \mathbf{d}_k - \rho\left[\mathbf{v}_{k+1} - \mathscr{D}(\mathbf{m}_{k+1} - \mathbf{z}_0)\right]$;
4. Update ρ according to Eq. (23);
5. Check convergence:
   if $f_1(\mathbf{m}_{k+1}) - f_1(\mathbf{m}_k) < \varepsilon$
   Stop with solution $\mathbf{m}_{k+1}$;
   else
   $k \leftarrow k+1$;
   end if

end for

Output: The optimal mask.

3.2. Source optimization problem

The cost function of source optimization also consists of a pattern fidelity term and one regularization term relating to the illumination source. For the former, the difference between the simulated circuit image and the desired image is still measured by the ℓ_2 norm. The penalty term is devised to achieve a design trade-off between the feasibility and manufacturability of using pixelated illumination in SMO technology, as used in our previous work [11]. Thus, the equivalent constrained source optimization problem is given by

$$\begin{aligned} \underset{\mathbf{s}}{\text{minimize}} \quad & f_2(\mathbf{s}) = \frac{\mu}{2}\left\|\mathbf{z} - \mathbf{z}_0\right\|_2^2 + \left\|\mathbf{v}'\right\|_1 \\ \text{subject to} \quad & \mathbf{v}' = \mathscr{D}(\mathbf{s}), \\ & \mathbf{s} \geq 0. \end{aligned} \tag{15}$$

The augmented Lagrangian of this equation can be represented by

$$L_\rho(\mathbf{s}, \mathbf{v}', \mathbf{d}') = \frac{\mu}{2}\left\|\mathbf{z} - \mathbf{z}_0\right\|_2^2 + \left\|\mathbf{v}'\right\|_1 - \mathbf{d}'^T\left[\mathbf{v}' - \mathscr{D}(\mathbf{s})\right] + \frac{\rho}{2}\left\|\mathbf{v}' - \mathscr{D}(\mathbf{s})\right\|_2^2. \tag{16}$$

As previously described, $L_\rho(\mathbf{s}, \mathbf{v}', \mathbf{d}')$ is firstly minimized with respect to $\mathbf{s}$ to find the solution of the original problem (15), and then we minimize $\mathbf{v}'$ and update $\mathbf{d}'$. As a consequence, the overall augmented Lagrangian algorithm for source optimization is composed of the following three sub-problems

$$\begin{aligned} \mathbf{s}_{k+1} = \arg\min_{\mathbf{s}} \quad & \frac{\mu}{2}\left\|\mathbf{z} - \mathbf{z}_0\right\|_2^2 - \mathbf{d}_k'^T\left[\mathbf{v}'_k - \mathscr{D}(\mathbf{s})\right] + \frac{\rho}{2}\left\|\mathbf{v}'_k - \mathscr{D}(\mathbf{s})\right\|_2^2 \\ \text{subject to} \quad & \mathbf{s} \geq 0, \end{aligned} \tag{17}$$

$$\mathbf{v}'_{k+1} = \arg\min_{\mathbf{v}'} \quad \left\|\mathbf{v}'\right\|_1 - \mathbf{d}_k'^T\left[\mathbf{v}' - \mathscr{D}(\mathbf{s}_{k+1})\right] + \frac{\rho}{2}\left\|\mathbf{v}' - \mathscr{D}(\mathbf{s}_{k+1})\right\|_2^2, \tag{18}$$

$$\mathbf{d}'_{k+1} = \mathbf{d}'_k - \rho\left[\mathbf{v}'_{k+1} - \mathscr{D}(\mathbf{s}_{k+1})\right]. \tag{19}$$

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

As with ALM for mask optimization, using L-BFGS-B requires the first derivative of $L_\rho(\mathbf{s},\mathbf{v}',\mathbf{d}')$ with respect to the source. Note that the upper bound of the source can be set to infinity. Thus, the gradient we need is

$$\frac{\partial L_\rho(\mathbf{s},\mathbf{v}',\mathbf{d}')}{\partial \mathbf{s}} = \mu\alpha\mathbf{I}_s^T\left[(\mathbf{z}-\mathbf{z}_0)\odot\mathbf{z}\odot(1-\mathbf{z})\right] + \mathscr{D}^T(\mathbf{d}'_k) - \rho\mathscr{D}^T(\mathbf{v}'_k) + \rho\mathscr{D}^T\mathscr{D}(\mathbf{s}). \quad (20)$$

The derivation is detailed in the Appendix. The solution of $\mathbf{v}'$ is similar to that of mask optimization, i.e.,

$$\mathbf{v}'_{k+1} = \max\left\{\left|\frac{\mathbf{d}'_k}{\rho} + \mathscr{D}(\mathbf{s}_{k+1})\right| - \frac{1}{\rho}, 0\right\}\odot \operatorname{sgn}\left(\frac{\mathbf{d}'_k}{\rho} + \mathscr{D}(\mathbf{s}_{k+1})\right). \quad (21)$$

Algorithm 2 lists the pseudo-code of the ALM for source optimization.

Table 2. Pseudo-code of source optimization procedure

Algorithm 2: ALM for source optimization problem

Input: Assign multiplier $\mathbf{d}'_1 = 0$ and the starting source s_1;
Choose convergence parameter ε; Set $\rho > 0$, $\tau > 1$, $0 < \alpha < 1$;

for k=1,2,... **do**

1. Solve the subproblem of $\mathbf{s}_{k+1}$ (17) using L-BFGS-B;
2. $\mathbf{v}'_{k+1} = \max\left\{\left|\mathbf{d}'_k/\rho + \mathscr{D}(\mathbf{s}_{k+1})\right| - 1/\rho, 0\right\}\odot \operatorname{sgn}\left(\mathbf{d}'_k/\rho + \mathscr{D}(\mathbf{s}_{k+1})\right)$;
3. Update the Lagrangian multiplier $\mathbf{d}'_{k+1} = \mathbf{d}'_k - \rho\left[\mathbf{v}'_{k+1} - \mathscr{D}(\mathbf{s}_{k+1})\right]$;
4. Update ρ according to Eq. (23);
5. Check convergence:
   if $f_2(\mathbf{s}_{k+1}) - f_2(\mathbf{s}_k) < \varepsilon$
   Stop with solution $\mathbf{s}_{k+1}$;
   else
   $k \leftarrow k+1$;
   end if

end for

Output: The optimal source.

On the basis of above description, we now generalize the conversion of the cost function in source optimization to an equivalent constrained one. Assuming the source optimization problem in which the cost function is the sum of the pattern fidelity term $f(\mathbf{s})$ and the regularization term $r(\mathbf{s})$, then the constrained problem is given by

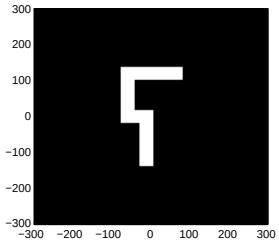
$$\begin{aligned} \underset{\mathbf{s}}{\text{minimize}} \quad & f(\mathbf{s}) + r_g(\mathbf{u}) \\ \text{subject to} \quad & \mathbf{u} = g(\mathbf{s}), \\ & \mathbf{s} \geq 0. \end{aligned} \quad (22)$$

Then ALM can be used to address Eq. (22). $r_g(\mathbf{u})$ is derived from $r(\mathbf{s})$. The ℓ_2 norm and total variation in this paper are specific conditions of $f(\mathbf{s})$ and $r(\mathbf{s})$, respectively. The generalized formulation for mask optimization is similar.

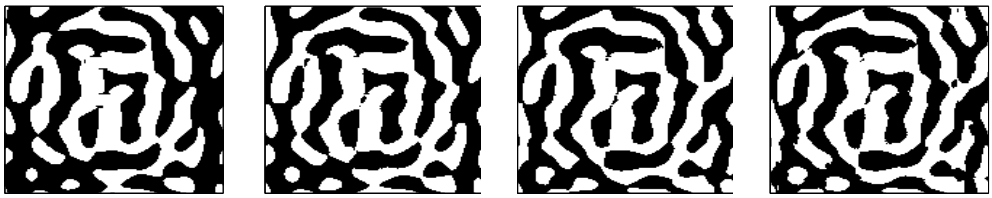
3.3. Parameters analysis

1. Choice of μ: To visualize the impact of different values of μ on the resulting image, we carry out a series of tests on a mask pattern with a critical dimension (CD) of 36 nm

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013



(a) Test pattern



(b) $\mu = 80$ (c) $\mu = 472$ (d) $\mu = 1000$ (e) $\mu = 13650$

(f) $PE = 123$ (g) $PE = 131$ (h) $PE = 71$ (i) $PE = 66$

Fig. 1. Simulation results of the test pattern with different choices of μ. Top row is the target image, and middle row presents the optimized masks and the corresponding outputs with pattern error (PE) are in the third row. The units of PE are in pixels.

(size 151×151, pixel resolution 4nm) by fixing the annular source ($0.7/0.9$ annulus). As illustrated in Fig. 1, large μ tends to deliver a pattern closer to the design, but small values yield relatively simpler masks. In our experiments, μ is set to 1000 for both mask and source optimization problems.

2. Choice of ρ: Rather than treating ρ as a fixed constant, we adopt the following update scheme

$$\rho = \begin{cases} \rho\ , & \text{if } \left\|\mathbf{v}_{k+1} - \mathscr{D}(\mathbf{m}_{k+1} - \mathbf{z}_0)\right\|_2 \leq \eta \\ \tau\rho, & \text{otherwise,} \end{cases} \tag{23}$$

where τ is the multiplication factor for updating ρ to be found empirically, and η is a constant to specify whether the current value of the penalty parameter is producing an acceptable level of constraint violation. This enables a faster rate of convergence as derived in [23]. Ideally, the condition $\frac{\rho}{2}\left\|\mathbf{v}_k - \mathscr{D}(\mathbf{m}_k - \mathbf{z}_0)\right\|_2^2$ should decrease as k increases [34]. However, if not, it can be forced to reduce by increasing its relative weight in the objective function. Hence, the update scheme of $\tau\rho$ guarantees the convergence of the proposed algorithm, and when the steady state is reached as k approaches infinity, ρ becomes a constant [36]. The update for ρ in source optimization follows a similar approach.

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

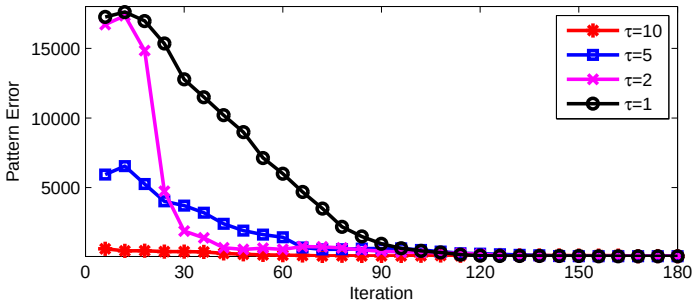


Fig. 2. Convergence profile of the proposed algorithm using different values of τ.

Empirically, a reasonable initial value of ρ typically lies in the range $[0.01, 10]$. Too large a value (in the order of 100) may miss the solution to the original problem, while too small a ρ will neglect the effect of the complexity penalty condition $\left\|\mathbf{v}_k - \mathscr{D}(\mathbf{m}_k - \mathbf{z}_0)\right\|_2^2$. The four colored curves in Fig. 2 show the convergence rate using different values of τ with Fig. 1(a) as the input pattern. We find that $\rho = 0.5$ and $\tau = 2$ are robust to most mask patterns.

4. Results

To evaluate the performance of the augmented Lagrangian algorithm for SMO, we first compare and analyze the simulation results in terms of pattern error and convergence rate, followed by a summary of the NILS, speed and complexity, and then we compare the process window for manufacturing conditions. Two target patterns are used, namely, a cross gate design and an alternatively arranged brick poly array, as shown in Figs. 3(a) and 3(b), respectively. Both are represented by a 151×151 matrix with a pixel size of $4\,\text{nm} \times 4\,\text{nm}$ and critical dimension (CD) of 36 nm. An annular illumination composed of 21×21 pixels with its inner annulus $\sigma_{in} = 0.7$ and outer annulus $\sigma_{out} = 0.9$ is adopted as the initial value for our source optimization. The parameters of the projection system are set to be $\lambda = 193\,\text{nm}$ and $NA = 1.35$. In the sigmoid function, t_r and α are equal to 0.3 and 85, respectively.

In order to evaluate the image fidelity, we compare the optimization results using our proposed ALM framework with an SMO solved by the nonlinear conjugate-gradient (CG) method [4], where the gate pattern in Fig. 3(a) is used as input. The nonlinear CG method is generally used to find the local minimum of a nonlinear function using its gradient. Here, we consider the results in terms of the pattern error at nominal condition, so process variations and regularization terms in [4] are not incorporated into the cost function of the SMO algorithm. Figures 4(a) – 4(c) respectively display the resulting source, the optimized mask and the printed image at nominal conditions using the proposed ALM. The corresponding results from CG method are given in the following row with the same structure. It is observed that our method can generate a similar output, while in addition, the regions around the corners and line-ends are better printed in Fig. 4(c), compared with that in Fig. 4(f). All the optimized sources are normalized by the maximum pixel intensity for better visualization.

As further evidence that the circuit pattern quality is indeed improved by ALM, another experiment is conducted with the periodic array of brick patterns in Fig. 3(b) as input, and we compute the SMO using both ALM and CG. Figure 5 presents the corresponding results in a way similar to the above. Comparing the two circuit images printed at the nominal process condition shown in Figs. 5(c) and 5(f), we observe that the former reduces the pattern errors

#184602 - $15.00 USD
Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

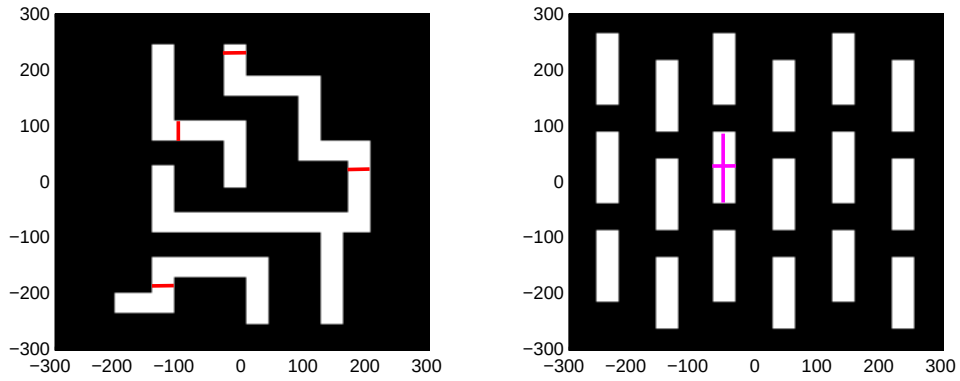


Fig. 3. Two test patterns used in experiments: (a) cross gate design and (b) brick poly array. Red and magenta lines mark the critical locations for measuring the process window of the two patterns, respectively.

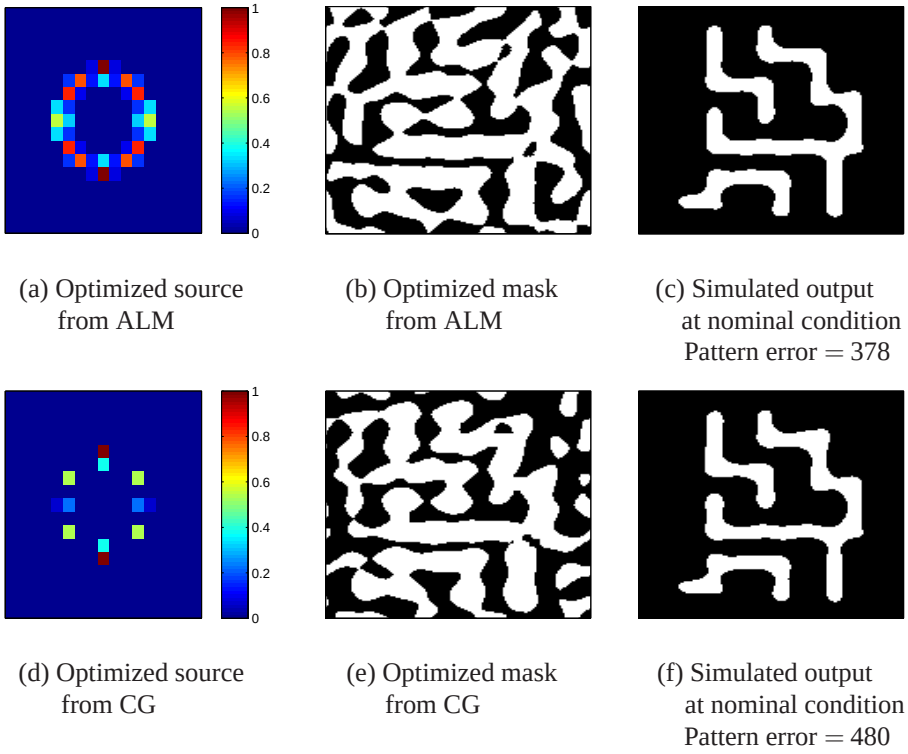
by 15% more than that of the latter. It is worth noting that the end regions of the poly array in Fig. 5(f) have more distortions, which signifies that our method has a better resolution enhancement capacity over such regions.

It should also be noted that SMO with the CG method is not capable of acquiring the best pattern fidelity in terms of the pattern error achieved by ALM even if it continues the iteration. Simulations conducted by ALM in Fig. 4(c) and Fig. 5(c), showing pattern error results of 378 and 410 under best focus for two test patterns respectively, are far better than the results in Fig. 4(f) and Fig. 5(f) (with pattern errors of 480 and 479, respectively). This is consistent with our observation in the corner areas of Figs. 4(c) and 4(f), as well as Figs. 5(c) and 5(f). This result is related to the fact that the CG method is only applicable to certain types of equations, where the Hessian matrix is positive definite. If it is not for a particular optimization problem, the CG method fails to work properly and the algorithm does not converge to the minimum. In other words, ALM can explore a larger solution space of the inverse problems than CG. The above two simulation results also affirm that the proposed ALM is suitable for printing both gate pattern and periodic array.

After evaluating the image quality of different algorithms, we can now assess the impact of the proposed augmented Lagrangian algorithm in terms of the convergence rate represented by the edge-placement error (EPE) versus the iteration number. EPE evaluates aerial image quality by measuring the difference between the ideal profile and the simulated edge placement. In the following analysis, one can see that the optimal source and mask in ALM can be found with less time than that with the CG method.

With the brick poly array as input, the blue and green lines in Fig. 6(b) are the center and near-end edges where we calculate EPE. We magnify these two regions in Figs. 6(a) and 6(c), where the cyan and magenta curves are the threshold contours of the aerial image after SMO calculation by using CG and ALM, respectively. The black curve is the target pattern contour. EPE evaluates the pattern fidelity by computing the distance between the output pattern contour and the desired pattern. During the optimization process, we observe that the EPE rapidly converges in the ALM for both the center and near-end lines. ALM reaches the optimal solution after 160 iterations, with a near 0 nm EPE for the middle regions. Meanwhile for CG, after undergoing the same time, it prints the edge marked by a magenta line with an EPE of 4.9 nm, and

#184602 - $15.00 USD
Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013



(a) Optimized source from ALM

(b) Optimized mask from ALM

(c) Simulated output at nominal condition Pattern error = 378

(d) Optimized source from CG

(e) Optimized mask from CG

(f) Simulated output at nominal condition Pattern error = 480

Fig. 4. Simulation results of the first test pattern.

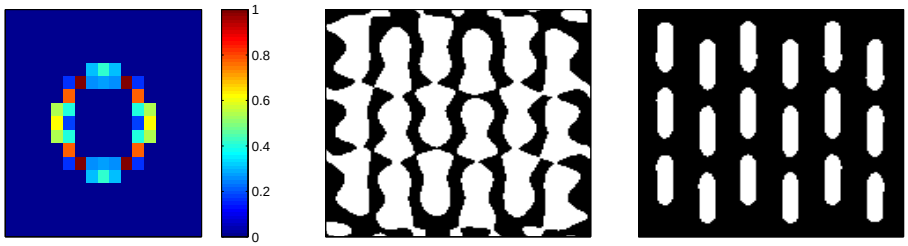
another 250 iterations are required to obtain the optimal solution. This is reasonable because the L-BFGS-B algorithm uses curvature information to take a more direct route as compared to CG. Moreover, ALM results show better overall EPE performance than CG results for both near-end and center regions, as illustrated in Fig. 6.

Furthermore, from both experiments, we notice that the optimized sources from CG contain a few isolated pixels. In contrast, our algorithm results in a manufacturing-friendly design by applying the complexity regularization term, adjusting source intensity more than its shape.

Table 3 summarizes the measurements of the pattern error, normalized image log slopes (NILS), speed and the smallest polygon for the two tests, where we compute with the ALM, as well as with an CG method and a level-set optimization scheme. Level-set method treats the mask design in lithography as an inverse mathematical problem, and solves it as a partial differential equation by a time-dependent model with finite difference schemes [14]. To compare the speed of the methods, in a way that is as independent as possible from the different stopping criteria, we run these methods until they reach similar pattern error values, where a shorter computation time indicates a faster speed. For both test patterns, when all three methods produce similar pattern errors, ALM takes the least time to achieve a better performance, exhibiting about 3 to 7 times convergence improvement, better NILS and reduced mask complexity.

Finally, Fig. 7 depicts the average exposure-defocus window comparison, which measures the sensitivity of pattern CD to defocus and exposure dose at critical locations. For the first pattern, the minimum feature size (also the width of the feature), line-end and corners are chosen as the critical regions, as marked by the red lines in Fig. 3(a). The magenta lines in Fig. 3(b) indicate that average PW is measured at all patterns width and length for the second test pat-

#184602 - $15.00 USD
Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013



(a) Optimized source from ALM

(b) Optimized mask from ALM

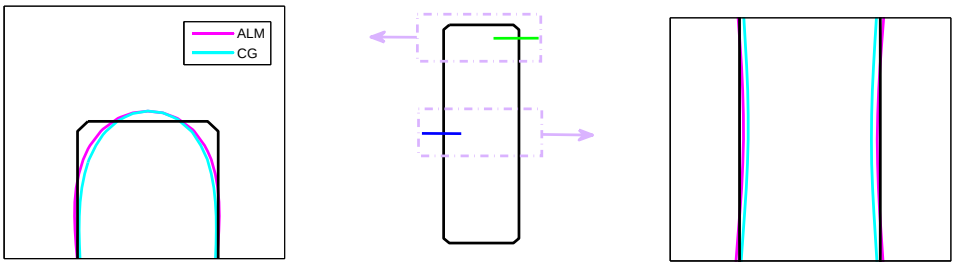
(c) Simulated output at nominal condition Pattern error = 410

(d) Optimized source from CG

(e) Optimized mask from CG

(f) Simulated output at nominal condition Pattern error = 479

Fig. 5. Simulation results of the second test pattern.



(a) (b) (c)

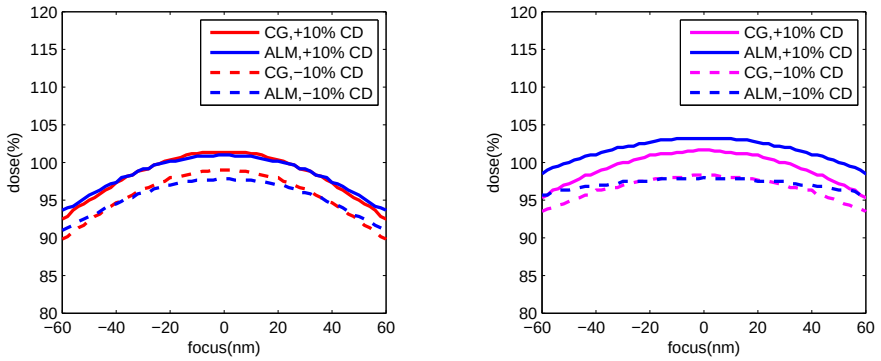
Fig. 6. Magnified aerial image threshold contours of (a) near-end regions (c) center regions. (b) is one magnified poly pattern with lines for EPE evaluation.

tern. We evaluate PW size by calculating how large the depth of focus (DoF) can be when the exposure latitude (EL) is fixed. Allowable maximum and minimum doses of the corresponding locations with linewidth change within 10% are plotted as different color curve pairs, indicating the proposed ALM and the conventional CG method for SMO, respectively. DoF is evaluated by checking the largest acceptable defocus range of an ellipse tangent with the color curve pairs at a particular dose. If we fix the EL condition at 5%, from Fig. 7(a), the blue and red curves quantitatively verify the similarity between process variation trends of the two methods. Our

#184602 - $15.00 USD
Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

Table 3. Comparison of performance and convergence rate

Test patterns	Methods	PE	NILS	Speed (sec)	Smallest polygon (pixel)
Cross gate design	CG [4]	480	0.55	84.7	3
	Level-set Method [14]	564	0.54	225.1	85
	Proposed ALM	472	0.57	29.3	62
Brick poly array	CG [4]	479	0.63	106.8	1
	Level-set Method [14]	556	0.59	199.8	21
	Proposed ALM	476	0.83	25.6	5



(a) CG DoF = 46 nm; ALM DoF = 56 nm (b) CG DoF = 60 nm; ALM DoF = 86 nm

Fig. 7. Comparison of average process window of (a) cross gate design and (b) brick poly array.

method allows a wider tolerance of process variations by increasing the DoF from 46 nm to 56 nm. A larger average PW can be observed in Fig. 7(b), where the proposed ALM increases the process capability by producing a 26 nm larger defocus range than the CG method, demonstrating enhanced variation robustness. Such results are reasonable because the improvement of PW is mainly displayed within the $\pm$30 nm defocus range, where the distortions from the process aberrations in this range can be mitigated by the extra ability of the high contrast images resulting from ALM. In further defocus positions, the output does not have more degradations.

5. Conclusions

In conclusion, an ALM-based inverse algorithm is developed for fast source and mask design in optical lithography. We investigate how the minimization problem is formulated into an equivalent constrained optimization problem, which is solved efficiently by applying an alternating scheme to decouple the minimization of the associated augmented Lagrangian function and by taking advantage of the computational advances in ALM. Advantages such as rapidly converging EPEs, high image fidelity and process robustness not only allow this approach to be a prime candidate for full-chip inverse lithography applications, but also afford algorithmic insights to how variable splitting and alternating minimization in the augmented Lagrangian framework solve large-scale unconstrained and constrained optimization problems.

#184602 - $15.00 USD Received 31 Jan 2013; accepted 8 Mar 2013; published 27 Mar 2013

A. Appendix: Gradients derivation

In the following we explain how to compute the derivatives of the augmented Lagrangian function in Eqs. (12) and (20), dropping index k for brevity. The first gradient of Eq. (8) with respect to the mask pattern is given by

$$\begin{aligned}\frac{\partial L_\rho(\mathbf{m},\mathbf{v},\mathbf{d})}{\partial \mathbf{m}} &= \frac{\mu}{2}\frac{\partial \left\|\mathbf{z}-\mathbf{z}_0\right\|_2^2}{\partial \mathbf{m}} - \mathbf{d}^T\frac{\partial\left[\mathbf{v}-\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right]}{\partial \mathbf{m}} + \frac{\rho}{2}\frac{\partial\left\|\mathbf{v}-\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right\|_2^2}{\partial \mathbf{m}} \\ &= \frac{\mu}{2}\left[2\alpha(\mathbf{z}-\mathbf{z}_0)\odot\mathbf{z}\odot(1-\mathbf{z})\odot\frac{\partial \mathbf{z}_a}{\partial \mathbf{m}}\right] + \mathscr{D}^T\mathbf{d} \\ &\quad + \frac{\rho}{2}\frac{\partial\left[\mathbf{v}-\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right]^T\left[\mathbf{v}-\mathscr{D}(\mathbf{m}-\mathbf{z}_0)\right]}{\partial \mathbf{m}} \\ &= \mu\alpha\mathrm{Re}\left\{\sum_{l=1}^{P}\lambda_l\left(\bar{\mathbf{H}}_l * \left[(\mathbf{z}-\mathbf{z}_0)\odot\mathbf{z}\odot(1-\mathbf{z})\odot(\tilde{\mathbf{H}}_l * \mathbf{m})^\dagger\right]\right)\right\} \\ &\quad + \mathscr{D}^T(\mathbf{d}) - \rho\mathscr{D}^T(\mathbf{v}) + \rho\mathscr{D}^T\mathscr{D}(\mathbf{m}) - \rho\mathscr{D}^T\mathscr{D}(\mathbf{z}_0). \end{aligned} \tag{24}$$

The analytical form of the partial gradients for the illumination source in Eq. (20) is

$$\begin{aligned}\frac{\partial L_\rho(\mathbf{s},\mathbf{v}',\mathbf{d}')}{\partial \mathbf{s}} &= \frac{\mu}{2}\frac{\partial \|\mathbf{z}-\mathbf{z}_0\|_2^2}{\partial \mathbf{s}} - \mathbf{d}'^T\frac{\partial\left[\mathbf{v}'-\mathscr{D}(\mathbf{s})\right]}{\partial \mathbf{s}} + \frac{\rho}{2}\frac{\partial\|\mathbf{v}'-\mathscr{D}(\mathbf{s})\|_2^2}{\partial \mathbf{s}} \\ &= \frac{\mu}{2}\left[2\alpha(\mathbf{z}-\mathbf{z}_0)\odot\mathbf{z}\odot(1-\mathbf{z})\odot\frac{\partial \mathbf{z}'_a}{\partial \mathbf{s}}\right] + \mathscr{D}^T\mathbf{d}' \\ &\quad + \frac{\rho}{2}\frac{\partial\left[-2\mathbf{v}'^T\mathscr{D}(\mathbf{s}) + \mathbf{s}^T\mathscr{D}^T\mathscr{D}(\mathbf{s})\right]}{\partial \mathbf{s}} \\ &= \mu\alpha\mathbf{I}_s^T\left[(\mathbf{z}-\mathbf{z}_0)\odot\mathbf{z}\odot(1-\mathbf{z})\right] + \mathscr{D}^T(\mathbf{d}') - \rho\mathscr{D}^T(\mathbf{v}') + \rho\mathscr{D}^T\mathscr{D}(\mathbf{s}). \end{aligned} \tag{25}$$

Acknowledgments

This work was supported in part by the Research Grants Council of the Hong Kong Special Administrative Region, China, under Project HKU 7134/08E, and by the UGC Areas of Excellence project Theory, Modeling, and Simulation of Emerging Electronics.