



Regularized level-set-based inverse lithography algorithm for IC mask synthesis

Zhen GENG[†], Zheng SHI, Xiao-lang YAN, Kai-sheng LUO

(Institute of VLSI Design, Zhejiang University, Hangzhou 310027, China)

[†]E-mail: gengzhen@vlsi.zju.edu.cn

Received Feb. 23, 2013; Revision accepted June 28, 2013; Crosschecked Sept. 16, 2013

Abstract: Inverse lithography technology (ILT) is one of the promising resolution enhancement techniques, as the advanced IC technology nodes still use the 193 nm light source. In ILT, optical proximity correction (OPC) is treated as an inverse imaging problem to find the optimal solution using a set of mathematical approaches. Among all the algorithms for ILT, the level-set-based ILT (LSB-ILT) is a feasible choice with good production in practice. However, the manufacturability of the optimized mask is one of the critical issues in ILT; that is, the topology of its result is usually too complicated to manufacture. We put forward a new algorithm with high pattern fidelity called regularized LSB-ILT implemented in partially coherent illumination (PCI), which has the advantage of reducing mask complexity by suppressing the isolated irregular holes and protrusions in the edges generated in the optimization process. A new regularization term named the Laplacian term is also proposed in the regularized LSB-ILT optimization process to further reduce mask complexity in contrast with the total variation (TV) term. Experimental results show that the new algorithm with the Laplacian term can reduce the complexity of mask by over 40% compared with the ordinary LSB-ILT.

Key words: Inverse lithography technology, Complexity, Level set, Regularization

doi:10.1631/jzus.C1300050

Document code: A

CLC number: TN47

1 Introduction

The process used for manufacturing integrated circuits (IC) includes a sequence of microlithographic steps in which patterns are formed by projection printing (Wong *et al.*, 2009). With the continuous downscaling of semiconductor critical dimensions (CD), the minimum feature size (e.g., 65 nm, 45 nm, 32 nm) of modern IC is much smaller than the light source wavelength (193 nm).

The CD on wafer is limited by the Rayleigh criterion (Wong, 2001), as shown by the following expression:

$$\text{Resolution} = k_1 \frac{\lambda}{\text{NA}}, \quad (1)$$

where λ is the wavelength of the light source, NA is

the numerical aperture of the projection system, and k_1 is the process-related factor. Hence, a smaller CD can be printed by increasing NA or decreasing the lithography wavelength. Although the immersion lithography system with a 193-nm light source has NA from 1 up to 1.3, the advanced technology (such as 45 nm) nodes need obviously the reduction of k_1 . Different resolution enhancement techniques (RETs) which control the amplitude, phase, and direction of the optical waves are applied to minimize the process-related factor k_1 (Schellenberg, 2004; Ma and Arce, 2010).

Optical proximity correction (OPC) is one of the most important RETs by adjusting the topology of the mask to make the printed pattern on the wafer as close to the desired pattern as possible (Lin *et al.*, 2011). In general, the OPC optimization methods can be divided into two classes: rule-based approaches and model-based approaches. Rule-based approaches are simple to implement, but they can compensate only

the warping in local features. Model-based approaches use mathematical models to represent the image formation process of the optical lithography system, and iteratively seek the optimal solution to improve the output pattern fidelity (Ma and Arce, 2010). In advanced technology (such as 45 nm and below) nodes, rule-based approaches do not work well; in contrast, inverse lithography technology (ILT), as a special case of model-based approaches, produces a much better result. ILT pixelizing the mask with equal size treats mask synthesis as an inverse problem, to find a proper topology of the mask. The mathematical description of ILT is shown as follows:

$$\text{contour} = \text{Litho}(\text{mask}), \quad (2)$$

$$\text{mask}^* = \text{Litho}^{-1}(z), \quad (3)$$

where contour is the simulation result on wafer, z represents the target patterns on wafer, and mask^* is the optimal mask calculated using the ILT algorithm (Li et al., 2012).

In recent years, various kinds of ILT technologies have been proposed. Granik (2004; 2006) described and compared solutions of inverse mask problems. Yu and Pan (2007) proposed a topological invariant pixel based OPC. Poonawala and Milanfar (2007a) formulated the mask synthesis problem using a continuous function optimization process and the gradient information to search the solution space. Shen et al. (2008) implemented a new inverse mask synthesis system using two-dimensional discrete cosine transform (DCT2) of the target mask. Pang et al. (2008) proposed an ILT algorithm based on the level set method. Shen et al. (2009) provided exact formulation and technical discussions in sufficient detail, and Shen et al. (2010) further proposed a statistical method which incorporates process variations into the inverse lithography problem. Jia and Lam (2010) treated mask optimization as a training process and adopted the stochastic gradient descent approach. Yu and Yu (2010) developed a gradient descent approach to investigate three different objective functions and their combinations. Ma et al. (2012a; 2012b) first developed robust pixelated gradient-based OPC and phase-shifting mask (PSM) optimization algorithms under a vector imaging model. Fig. 1 gives an example to show the mask layout generated by ILT.

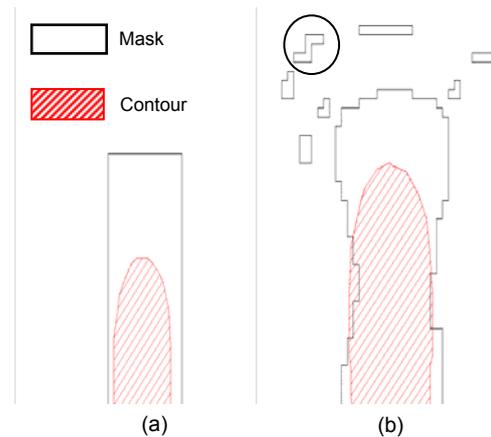


Fig. 1 A comparison between the original design and the ILT optimized mask with their simulated contours

In (a), the rectangle is the original mask pattern, and the inner shaded contour is the simulation result of the original mask. In (b), all blank polygons form the optimized mask result and the inner shaded contour is the simulated wafer pattern of the optimized mask

There are many irregular patterns, such as the circled one, in the optimized mask result. With this kind of complicated topology, manufacturing the mask is costly.

No matter which ILT algorithm is adopted, irregular patterns will be generated, resulting in great mask complexity. To enhance the manufacturability of the mask generated by ILT, different methods have been developed. Poonawala and Milanfar (2007b) employed the total variation (TV) penalty function to suppress isolated perturbations and protrusions of the mask. Ma and Arce (2008) introduced an effective detail-reduction approach referred to as ‘wavelet penalty’, and Ma and Arce (2011) proposed a mask rule check (MRC) penalty to obtain more desirable manufacturability characteristics for the optimized mask. Jia et al. (2009) put forward a scheme to explicitly discuss this problem. However, all these methods are implemented in the gradient-based ILT (GB-ILT).

The level-set-based ILT (LSB-ILT) represents the mask as a 2D level-set function and the representation allows contours to merge, break, appear, or disappear, in a consistent, mathematical representation (Pang et al., 2008). However, few studies were concerned about the complexity of the mask generated by LSB-ILT. To reduce the complexity of the mask, we propose a new algorithm named regularized

level-set-based ILT (RLSB-ILT) by changing the cost function. In addition, we put forward a new regularization term, namely the Laplacian term, to reduce the mask complexity in contrast with the TV term. The new algorithm is implemented in a partially coherent illumination (PCI) condition and can be applied to real manufacturing processes.

2 Level set method and regularization term

2.1 Introduction to the level set method

The level set method, proposed by Osher and Sethian (1988), is a numerical technique for tracking interfaces and shapes. The advantage of this method is that one can perform numerical computations involving curves and surfaces on a fixed Cartesian grid without having to parameterize these objects.

Santosa (1996) developed the level-set approach to solve inverse problems. An inverse problem can be posed as

$$\mathbf{g} = A(\mathbf{u}) + \mathbf{n}, \quad (4)$$

$$W(\mathbf{u}) = \frac{1}{2} \|A(\mathbf{u}) - \mathbf{g}\|^2, \quad (5)$$

where \mathbf{g} , \mathbf{u} , and \mathbf{n} are variables defined in \mathbb{R}^2 , $\|\cdot\|$ is the L_2 norm, \mathbf{g} is the given data, and \mathbf{u} represents the model parameters. The function A in Eq. (4), the forward map, is a map from model parameters to data. \mathbf{n} represents the distance between the predicted data $A(\mathbf{u})$ and the given data \mathbf{g} . Eq. (5) is also called the decreasing function as a metric of \mathbf{n} . An inverse problem can be described as seeking the \mathbf{u} which minimizes $W(\mathbf{u})$.

In the level-set approach, \mathbf{u} is described by the level-set function φ as follows:

$$\varphi(\mathbf{x}) = \begin{cases} \varphi_{\text{int}}, & \mathbf{x} \in \mathbf{u}^-, \\ 0, & \mathbf{x} \in \partial\mathbf{u}, \\ \varphi_{\text{ext}}, & \mathbf{x} \in \mathbf{u}^+, \end{cases} \quad (6)$$

where \mathbf{x} stands for the coordinate (x, y) of one point in the plane and $\partial\mathbf{u}$ is the zero level set of φ , $\varphi(\mathbf{x})=0$, namely the boundary of \mathbf{u} . \mathbf{u}^- and \mathbf{u}^+ mean the inner part and outer part of \mathbf{u} , respectively. In optimization, the zero level set moves with a speed function V (Fig. 2).

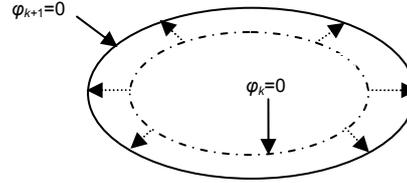


Fig. 2 Evolution example of the level-set function

The dotted arrows stand for the speed function V , the inner dotted ellipse is the k th zero level set curve, and the outer ellipse is the result of moving the k th one, the $(k+1)$ th zero level set curve

To formulate this process in Fig. 2, we can deduce the following function:

$$\varphi_{k+1}(\mathbf{x}) = \varphi_k(\mathbf{x}) + \frac{\partial\varphi}{\partial t} \Delta t, \quad (7)$$

where $\partial\varphi/\partial t$ is the speed function V . To obtain the solution of the speed function, Santosa (1996) deduced the level set equation as follows:

$$\frac{\partial\varphi}{\partial t} + [J(\mathbf{u})^T (A(\mathbf{u}) - \mathbf{g})] |\nabla\varphi| = 0, \quad (8)$$

where $J(\mathbf{u})^T$ is the Jacobian of $A(\mathbf{u})$ at \mathbf{u} and $\nabla\varphi$ is the gradient of φ . Then we can update $\varphi(\mathbf{x})$ according to Eq. (7) generating a sequence of level set φ_k and the correspondence $\partial\mathbf{u}_k$, until the desired solution $\mathbf{u}_{\text{desired}}$ is reached.

2.2 Forward lithography process

The function of forward lithography process F consists mainly of two models, a lithography model and a resist development model. The practical partially coherent imaging model can be decomposed into a sum-of-coherent-systems model (SOCS) (Cobb et al., 1996) based on Hopkins' imaging equation (Hopkins, 1953). The intensity of point (x, y) with the mask environment \mathbf{M} can be calculated by (Cobb and Zakhor, 1995)

$$\begin{aligned} I(x, y; \mathbf{M}) &= \sum_{i=1}^N \lambda_i |\mathbf{K}_i \otimes \mathbf{M}|^2 \\ &= \sum_{i=1}^N \lambda_i \left[|\text{Re}(\mathbf{K}_i) \otimes \mathbf{M}|^2 + |\text{Im}(\mathbf{K}_i) \otimes \mathbf{M}|^2 \right], \end{aligned} \quad (9)$$

where λ_i represents the i th weight value, \mathbf{K}_i is the i th lithography kernel, $\text{Re}(\mathbf{K}_i)$ and $\text{Im}(\mathbf{K}_i)$ are the real and imaginary parts of \mathbf{K}_i , respectively, \mathbf{M} is the mask matrix, and ‘ \otimes ’ is the convolution operator.

The constant threshold resist (CTR) model is described by the sigmoid function in Fig. 3, whose input is light intensity $\mathbf{I}(x, y)$ and the output of the sigmoid function indicates the resist thickness.

$$\text{sig}(\mathbf{I}) = \frac{1}{1 + \exp[-a(\mathbf{I} - t_r)]}, \quad (10)$$

where a is the steepness of the sigmoid and t_r is the resist image threshold.

We can obtain F , a nonlinear function, by combining the lithography model and the resist model in Eq. (10):

$$F(\mathbf{M}) = \text{sig}(\mathbf{I}(\mathbf{M})), \quad (11)$$

where $F(\mathbf{M})$ is the forward lithography process.

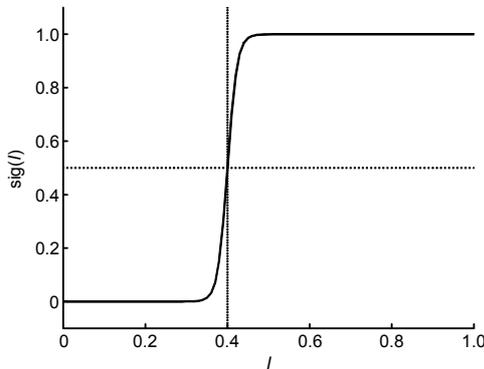


Fig. 3 Sigmoid function $\text{sig}(\mathbf{I})=1/\{1+\exp[-85(\mathbf{I}-0.4)]\}$

2.3 Regularization method

Many shape optimization problems, especially those arising from the inverse problem, are ill-posed. There exists no unique solution. From the multiple solutions we need to choose a solution that is more favorable to us. Regularization methods can be selected to impose desirable properties on the solutions.

In general, we adopt the approach that adds the regularization term to the cost function as follows:

$$\min_{\chi} [\text{cost}(\chi) + \alpha \cdot \text{reg}(\chi)], \quad (12)$$

where $\text{cost}(\chi)$ means the cost function, α is the regularization parameter, usually very small, and $\text{reg}(\chi)$ is the regularization function. The regularization term helps direct the unknown χ towards the solution space that we expect. In this study, we propose a new regularization term named the Laplacian term, which can further reduce the mask complexity in a shorter time, and select the TV regularization term as a contrast.

3 RLSB-ILT algorithm

In general, an ILT algorithm is aimed to minimize the cost function $\text{cost}(\mathbf{M})$:

$$\begin{aligned} \text{cost}(\mathbf{M}) &= \sum_{j=1}^m \sum_{i=1}^n [\text{sig}(\mathbf{I}(i, j)) - \mathbf{T}(i, j)]^2 \\ &= \sum_{j=1}^m \sum_{i=1}^n [F(\mathbf{M}(i, j)) - \mathbf{T}(i, j)]^2, \end{aligned} \quad (13)$$

where \mathbf{M} , \mathbf{I} , $\mathbf{T} \in \mathbb{R}^{m \times n}$, \mathbf{M} is the mask transmission matrix, \mathbf{I} denotes the image intensity distribution, and \mathbf{T} is the matrix representing the designed pattern. At the beginning of the new algorithm, matrices \mathbf{M} and \mathbf{T} are the same.

The overall flow of the new algorithm is as illustrated in Fig. 4. The partially coherent illumination (PCI) has been shown to improve the theoretical resolution limit in lithography (Ma and Arce, 2008). The speed function (SF) consists of two parts, namely Jacobian speed and regularization speed, which are used in the next step to control the evolution of ϕ . The value of the cost function is set as the criterion to determine whether the mask is optimized enough.

3.1 Time-dependent model

In level set methods, we introduce a function $B(\phi)$ to describe the relationship between level-set matrix ϕ and mask matrix \mathbf{M} , defined as

$$M(\mathbf{x}) = B(\phi) = \begin{cases} 1, & \mathbf{x} : \phi(\mathbf{x}) > 0, \\ 0, & \mathbf{x} : \phi(\mathbf{x}) \leq 0, \end{cases} \quad (14)$$

$$\phi(\mathbf{x}) = \begin{cases} -d(\mathbf{x}), & \mathbf{x} \in \mathbf{M}^-, \\ 0, & \mathbf{x} \in \partial\mathbf{M}, \\ d(\mathbf{x}), & \mathbf{x} \in \mathbf{M}^+, \end{cases} \quad (15)$$

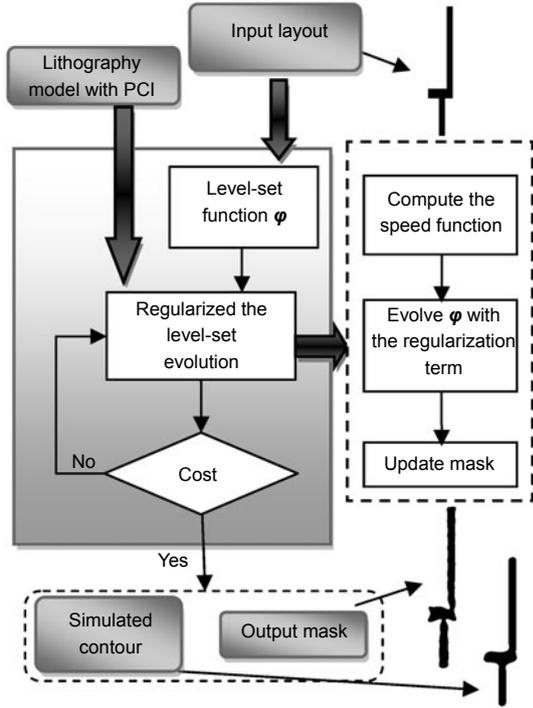


Fig. 4 Overall flow of regularized level-set based ILT

where \mathbf{x} represents the spatial coordinates (x, y) , $d(\mathbf{x})$ represents the distance between pixel \mathbf{x} and the boundary of \mathbf{M} . $\partial\mathbf{M}$ is the boundary of mask pattern \mathbf{M} . The zero level set of φ , $\varphi(\mathbf{x})=0$, is defined on the boundary of mask pattern. \mathbf{M}^- and \mathbf{M}^+ represent the inner part and outer part of the mask, respectively. For the inner part, the value of function $\varphi(\mathbf{x})$ is negative, namely $-d(\mathbf{x})$; for the outer part, the value is positive, namely $d(\mathbf{x})$.

Unlike other ILTs, level-set based ILT (LSB-ILT) optimizes $\varphi(\mathbf{x})$ from the evolution equation instead of \mathbf{M} . Shen *et al.* (2009) gave a detailed derivation optimizing the $\varphi(\mathbf{x})$ using LSB-ILT in a coherent image system. In the real world, however, the imaging systems are partially coherent. In our new algorithm, we apply a sum-of-coherent-systems method (Cobb *et al.*, 1996) to approximate the Hopkins partial coherence model.

From Eq. (8), we can derive the time-dependent evolution equation in the partially coherent imaging system:

$$\frac{\partial\varphi}{\partial t} = -\alpha(\mathbf{M}, t)|\nabla\varphi|, \quad (16)$$

where $\alpha(\mathbf{M}, t)$ is the Jacobian of the decreasing func-

tion at \mathbf{M} , namely Jacobian speed. In our ILT problem, the decreasing function is the cost function (13). Thus, the Jacobian part can be calculated as follows:

$$\begin{aligned} \alpha(\mathbf{M}) &= \frac{\partial}{\partial\mathbf{M}}(\text{sig}(\mathbf{I}) - \mathbf{T}_0)^2 \\ &= \frac{\partial}{\partial\mathbf{M}} \left[\text{sig} \left(\sum_{i=1}^N \lambda_i |\mathbf{K}_i \otimes \mathbf{M}|^2 \right) - \mathbf{T}_0 \right]^2 \\ &= \frac{\partial}{\partial\mathbf{M}} \left[\text{sig} \left(\sum_{i=1}^N \lambda_i \left((\text{Re}(\mathbf{K}_i) \otimes \mathbf{M})^2 + (\text{Im}(\mathbf{K}_i) \otimes \mathbf{M})^2 \right) \right) - \mathbf{T}_0 \right]^2 \\ &= \sum_{i=1}^N \lambda_i \cdot a \left[\text{Re}(\mathbf{K}_i) \otimes (\mathbf{T}_0 - \text{sig}(\mathbf{I})) \oplus \text{sig}(\mathbf{I}) \right. \\ &\quad \oplus (1 - \text{sig}(\mathbf{I})) \oplus (\text{Re}(\mathbf{K}_i) \otimes \mathbf{M}) \\ &\quad \left. + \text{Im}(\mathbf{K}_i) \otimes (\mathbf{T}_0 - \text{sig}(\mathbf{I})) \oplus \text{sig}(\mathbf{I}) \right. \\ &\quad \left. \oplus (1 - \text{sig}(\mathbf{I})) \oplus (\text{Im}(\mathbf{K}_i) \otimes \mathbf{M}) \right], \end{aligned} \quad (17)$$

where \mathbf{T}_0 represents the designed pattern and ‘ \oplus ’ means the Hadamard product (element-by-element multiplication) of two matrices.

3.2 Overall implementation of the RLSB-ILT algorithm

Using the ordinary LSB-ILT algorithm, many unwanted patterns will be generated in the mask, which will increase the cost of mask manufacturing. To suppress these patterns, we adopt the penalty method by adding a regularization function to the cost function (Marquina and Osher, 2000) and propose a new algorithm, namely the RLSB-ILT algorithm.

In the new algorithm, the regularization term is defined as $\text{Reg}(\partial\varphi/\partial x, \partial\varphi/\partial y)$ to control the regularity of φ . The problem can be formulated as follows:

$$\varphi = \arg \min_{\varphi} \left[\text{cost}(\mathbf{B}(\varphi)) + \lambda_{\text{Reg}} \text{cost}_{\text{Reg}} \left(\frac{\partial\varphi}{\partial x}, \frac{\partial\varphi}{\partial y} \right) \right], \quad (18)$$

$$\text{cost}_{\text{Reg}} \left(\frac{\partial\varphi}{\partial x}, \frac{\partial\varphi}{\partial y} \right) = \int_{\Omega} \text{Reg} \left(\frac{\partial\varphi}{\partial x}, \frac{\partial\varphi}{\partial y} \right) dx, \quad (19)$$

where λ_{Reg} is a user-defined weight of the regularization function cost_{Reg} and $\Omega = \{(x, y) | 0 \leq x \leq m, 0 \leq y \leq n\}$ means the area of the level-set matrix φ . By expanding the Euler-Lagrange equation of Eq. (18), we can conclude the regularized level-set evolution equation as follows:

$$\begin{aligned} \frac{\partial \phi}{\partial t} &= -\alpha(\mathbf{M}, t) |\nabla \phi| + \lambda_{\text{Reg}} \beta(\phi, t) |\nabla \phi| \\ &= -\alpha(\mathbf{M}, t) |\nabla \phi| + \lambda_{\text{Reg}} \left(\frac{\partial}{\partial x} \text{Reg}_u + \frac{\partial}{\partial y} \text{Reg}_v \right) |\nabla \phi|, \end{aligned} \quad (20)$$

where u means $\partial \phi / \partial x$ and v means $\partial \phi / \partial y$, Reg_u and Reg_v represent the partial derivatives. $\alpha(\mathbf{M}, t)$ can be calculated according to Eq. (17). $\beta(\phi, t)$ represents the regularization speed function. Different regularization functions can be used to compute $\beta(\phi, t)$.

Our new algorithm procedure is given as the following:

Algorithm 1 RLSB-ILT algorithm

Input: the mask that needs to be optimized (\mathbf{M}), desired pattern (z_0), optical kernels, and resist parameters.

Output: updateMask.

```

1 updateMask= $\mathbf{M}$ 
2 updateLevelimage= $\phi$ 
3 ITER=0
4 repeat
5   Compute Jacobian speed  $\alpha(\mathbf{M}, t)$ 
6   Compute regularization speed  $\beta(\phi, t)$ 
7   Compute SF= $-\alpha(\mathbf{M}, t) + \lambda \cdot \beta(\phi, t)$ 
8   delta $\phi \leftarrow$  upwind(SF,  $\phi$ )
// upwind is the numerical method to evaluate the
// spatial derivatives of  $\phi$  (Shen et al., 2009)
9   update $\phi =$  update $\phi + dt \cdot$  delta $\phi$ 
10  updateMask $\leftarrow$  B( $\phi$ )
11  Compute costFunction
12  ITER+=1
13 until costfunction < setnumber
```

3.3 Regularization term

In general, many studies on the level set method refer to the TV term as the regularization function. The TV term does not penalize discontinuities in ϕ , and thus allows us to recover the edges of the original image (Marquina and Osher, 2000). The TV regularization function is expressed as

$$\text{Reg}_{\text{TV}} \left(\frac{\partial \phi}{\partial x}, \frac{\partial \phi}{\partial y} \right) = \sqrt{\left(\frac{\partial \phi}{\partial x} \right)^2 + \left(\frac{\partial \phi}{\partial y} \right)^2}. \quad (21)$$

The regularization speed function $\beta(\phi, t)$ of the TV term is expressed as follows:

$$\beta(\phi, t) = \frac{\partial}{\partial x} \text{Reg}_u + \frac{\partial}{\partial y} \text{Reg}_v = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}. \quad (22)$$

$\nabla \cdot (\nabla \phi / |\nabla \phi|)$ is usually called the mean curvature.

We propose a new term called the Laplacian regularization term, which can also be used to efficiently reduce the irregular pattern generated in the mask. For the Laplacian regularization function,

$$\text{Reg}_{\text{Laplacian}} \left(\frac{\partial \phi}{\partial x}, \frac{\partial \phi}{\partial y} \right) = \left(\frac{\partial \phi}{\partial x} \right)^2 + \left(\frac{\partial \phi}{\partial y} \right)^2. \quad (23)$$

Similarly, the regularization speed function, namely $\beta(\phi, t)$, is expressed as

$$\beta(\phi, t) = \frac{\partial}{\partial x} \text{Reg}_u + \frac{\partial}{\partial y} \text{Reg}_v = \nabla \cdot (\nabla \phi) = \Delta \phi. \quad (24)$$

4 Experiments and comparison

We implemented the regularized level-set-based algorithm using Matlab, and adopted an advanced node lithography model which includes $\lambda=193$ nm, NA=1.2, with quad-crescent illumination. The test patterns were taken mainly from a 40 nm layout design. All computations were performed on a Dell PowerEdge R610 (Xeon 2.8 GHz and 32 GB memory) workstation.

To verify the quality and efficiency of our new algorithm, different test patterns have been used, which are classified as three groups:

1. Special layout pattern (SLP): SLP consists of five special layout patterns. Three of them are 36 nm layout examples to demonstrate the convergence efficiency of our algorithm and the other two patterns are from the 40 nm layout design.

2. Medium layout pattern (MLP): MLP consists of five medium area layouts from the 40 nm design layout.

3. Large layout pattern (LLP): LLP consists of five large area layouts from the 40 nm design layout.

Because the experimental layouts were with a small CD, the pixel grid was set to the appropriate size of 3 nm \times 3 nm. The value of the cost function in Eq. (13) was used as a metric of contour fidelity. Besides, the mask fabricating time and cost were

proportional to the number of rectangles in the fractured mask pattern. We adopted the total number of rectangles as a metric of mask complexity, which can be expressed as (Ma and Li, 2011)

$$\varepsilon = \sum_{i=1}^K \left(\frac{3}{4} \text{concave}_i + \frac{1}{4} \text{convex}_i \right), \quad (25)$$

where ε is the total number of rectangles in the test pattern, $\text{concave}_i \in \mathbb{N}^+$ denotes the number of all the concave vertices of one polygon, $\text{convex}_i \in \mathbb{N}^+$ denotes the number of all the convex vertices of one polygon, and K is the number of all polygons in the test pattern (Ma and Li, 2011).

To measure the algorithm's strength for reducing mask complexity, we define the complexity reduction rate η as follows:

$$\eta = \frac{\varepsilon_o - \varepsilon_r}{\varepsilon_o} \times 100\%, \quad (26)$$

where ε_o means the ε using the ordinary LSB-ILT algorithm and ε_r means the ε using the RLSB-ILT algorithm. The simulation results are presented in Table 1.

The initial cost calculated as $\text{cost}(\mathbf{T})$ means

contour fidelity before mask optimization. \mathbf{T} is the design layout pattern. The final cost calculated as $\text{cost}(\text{OptM})$ stands for contour fidelity after mask optimization, where OptM means the optimized mask.

For every test pattern, the final costs with three algorithms were set to the same level, as Shen *et al.* (2009) did, to compare the complexity of the mask in the same condition.

As a result, our new algorithm showed high complexity reduction efficiency. The average reduction rate η with the Laplacian term was 41.08%, and the average reduction rate η with the TV term was 33.97%.

Figs. 5a–5c show the masks optimized using the LSB-ILT algorithm, the RLSB-ILT algorithm with the Laplacian term, and the RLSB-ILT algorithm with the TV term, respectively.

The simulated contours on wafer are shown in Fig. 6. Fig. 6a is the design layout. Figs. 6b–6d are the simulated contours on wafer using the LSB-ILT algorithm, the RLSB-ILT algorithm with the Laplacian term, and the RLSB-ILT algorithm with the TV term, respectively. The edge placement errors (EPEs) between the simulated contours and the design layout of the three contours are all within the EPE accuracy range of 6 nm.

Table 1 Performance comparison between the ordinary LSB-ILT algorithm and the RLSB-ILT algorithm with the Laplacian term or TV term

Group	Test target	Initial cost	Final cost	ε metric ^a	ε metric ^b	ε metric ^c	Rate η^b (%)	Rate η^c (%)
SLP	SGL1*	6648	2598±5	252	138	170	45.24	32.54
	SGL2*	5891	3088±5	292	142	161	51.40	44.86
	NOR2*	9596	3799±5	298	216	238	27.51	20.13
	OR1	7039	4229±5	186	122	145	34.41	22.04
	INV0	6675	3519±5	168	93	101	44.64	39.88
MLP	AND2V	11 978	4596±5	588	323	350	45.07	40.48
	NAND2H	16 091	5994±5	527	378	418	28.27	20.68
	NOR3	11 881	5199±5	794	410	451	48.36	43.20
	OR2H	10 670	5249±5	496	340	372	31.45	25.00
	OR2V	10 776	5399±5	658	360	400	45.29	39.21
LLP	NOR4H	39 740	13 397±5	3733	1885	2317	49.50	37.93
	AND4H	42 839	14 494±5	4233	2737	2918	35.34	31.07
	DQ4V	37 882	12 779±5	2599	1620	1806	37.67	30.51
	NAND4V	39 823	13 417±5	3385	1951	2099	42.36	37.99
	OR4H	42 473	14 318±5	3645	1835	2039	49.66	44.06

The first three patterns with ‘*’ are the 36 nm layout pattern. ^a Using the ordinary LSB-ILT algorithm; ^b using the RLSB-ILT algorithm with the Laplacian regularization function; ^c using the RLSB-ILT algorithm with the TV regularization function

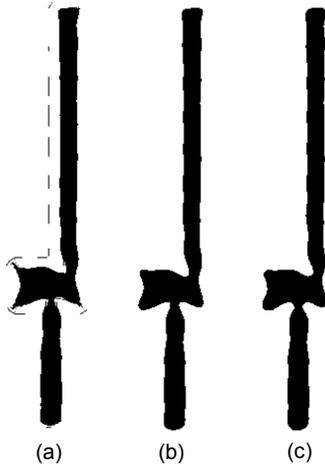


Fig. 5 Optimized masks with pattern ORI

(a)–(c) are the optimized masks obtained using the LSB-ILT algorithm, the RLSB-ILT algorithm with the Laplacian term, and the RLSB-ILT algorithm with the TV term, respectively

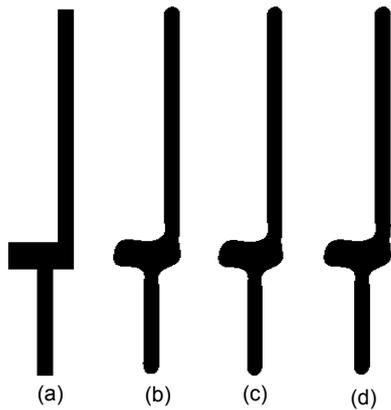


Fig. 6 Simulated contours with pattern ORI

(a) is the design layout; (b)–(d) are the simulated contours on wafer obtained using the LSB-ILT algorithm, the RLSB-ILT algorithm with the Laplacian term, and the RLSB-ILT algorithm with the TV term, respectively

Since the running time varies for different patterns, the simulation time of the ordinary LSB-ILT algorithm was set as the comparison basis to which the simulation times with the other two algorithms were normalized. Fig. 7 shows the relative simulation time using three methods, namely the ordinary LSB-ILT, RLSB-ILT with the Laplacian penalty term, and RLSB-ILT with the TV penalty term, among 15 experiment patterns. The average simulation time using RLSB-ILT with the Laplacian penalty term was 1.191 and the average simulation time using RLSB-ILT with the TV penalty term was 1.361.

As a result, in the partially coherent illumination condition, the ordinary LSB-ILT algorithm generated many irregular patterns in the optimized mask. This greatly increased the mask complexity. Our new algorithm can suppress these unwanted patterns from being generated and reduce the mask complexity obviously. Besides, for the two different regularization terms, the Laplacian term showed much better performance than the TV term. The Laplacian term was 7.11% better than the TV term in terms of the complexity reduction rate, and its optimization time was only 87.51% of that of the TV term.

5 Conclusions

In this paper, we propose a new ILT algorithm called the regularized level-set-based inverse lithography algorithm. Our new algorithm is based on the partially coherent imaging model which is the lithography condition applied on the practical

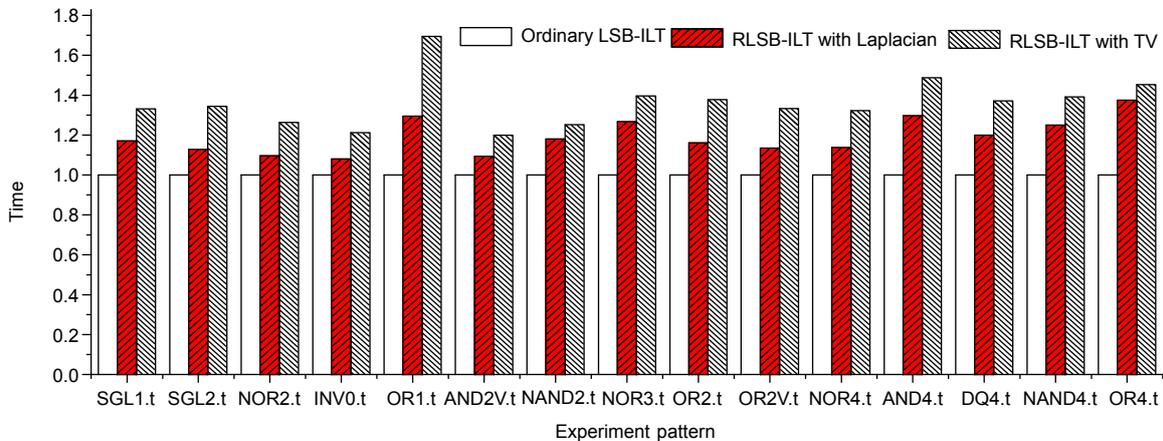


Fig. 7 Comparison of the simulation time of 15 test patterns between the ordinary LSB-ILT algorithm and the RLSB-ILT algorithm with the Laplacian term or TV term

manufacturing process. By adding the regularization function in the optimization process, our new algorithm shows better performance in reducing mask complexity. Furthermore, we propose a new regularization term named the Laplacian penalty term. Experimental results illustrate that the RLSB-ILT algorithm with the Laplacian term performs better than it with the TV term, with a higher complexity reduction rate (7.11% more than TV's) and a shorter simulation time (87.51% of TV's).

Reference

- Cobb, N.B., Zakhor, A., 1995. Fast sparse aerial image calculation for OPC. *SPIE*, **2621**:534-545. [doi:10.1117/12.228208]
- Cobb, N.B., Zakhor, A., Miloslavsky, E.A., 1996. Mathematical and CAD framework for proximity correction. *SPIE*, **2726**:208-222. [doi:10.1117/12.240907]
- Granik, Y., 2004. Solving inverse problems of optical lithography. *SPIE*, **5754**:506. [doi:10.1117/12.600141]
- Granik, Y., 2006. Fast pixel-based mask optimization for inverse lithography. *J. Micro/Nanolith. MEMS MOEMS*, **5**(4):043002. [doi:10.1117/1.2399537]
- Hopkins, H.H., 1953. On the diffraction theory of optical images. *Proc. R. Soc. A*, **217**(1130):408-432. [doi:10.1098/rspa.1953.0071]
- Jia, N.N., Lam, E.Y., 2010. Machine learning for inverse lithography: using stochastic gradient descent for robust photomask synthesis. *J. Opt.*, **12**(4):045601. [doi:10.1088/2040-8978/12/4/045601]
- Jia, N.N., Wong, A.K., Lam, E.Y., 2009. Regularization of inverse photomask synthesis to enhance manufacturability. *SPIE*, **7520**:75200E. [doi:10.1117/12.837512]
- Li, Y.H., Shi, Z., Geng, Z., Yang, Y.W., Yan, X.L., 2012. A new algorithm of inverse lithography technology for mask complexity reduction. *J. Semicond.*, **33**(4):045009. [doi:10.1088/1674-4926/33/4/045009]
- Lin, B., Yan, X.L., Shi, Z., Yang, Y.W., 2011. A sparse matrix model-based optical proximity correction algorithm with model-based mapping between segments and control sites. *J. Zhejiang Univ-Sci. C (Comput. & Electron.)*, **12**(5):436-442. [doi:10.1631/jzus.C1000219]
- Ma, X., Arce, G.R., 2008. Binary mask optimization for inverse lithography with partially coherent illumination. *J. Opt. Soc. Am. A*, **25**(12):2960-2970. [doi:10.1364/JOSAA.25.002960]
- Ma, X., Arce, G.R., 2010. Computational Lithography. Wiley Series in Pure and Applied Optics. Wiley & Sons, Hoboken, New Jersey, p.11. [doi:10.1002/9780470618943]
- Ma, X., Arce, G.R., 2011. Pixel-based OPC optimization based on conjugate gradient. *Opt. Expr.*, **19**(3):2165-2180. [doi:10.1364/OE.19.002165]
- Ma, X., Li, Y.Q., 2011. Resolution enhancement optimization methods in optical lithography with improved manufacturability. *J. Micro/Nanolith. MEMS MOEMS*, **10**(2):023009. [doi:10.1117/1.3590252]
- Ma, X., Li, Y.Q., Dong, L.S., 2012a. Mask optimization approaches in optical lithography based on a vector imaging model. *J. Opt. Soc. Am. A*, **29**(7):1300-1312. [doi:10.1364/JOSAA.29.001300]
- Ma, X., Li, Y.Q., Guo, X.J., Dong, L.S., Arce, G.R., 2012b. Vectorial mask optimization methods for robust optical lithography. *J. Micro/Nanolith. MEMS MOEMS*, **11**(4):043008. [doi:10.1117/1.JMM.11.4.043008]
- Marquina, A., Osher, S., 2000. Explicit algorithms for a new time dependent model based on level set motion for nonlinear deblurring and noise removal. *SIAM J. Sci. Comput.*, **22**(2):387-405. [doi:10.1137/S1064827599351751]
- Osher, S., Sethian, J.A., 1988. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.*, **79**(1):12-49. [doi:10.1016/0021-9991(88)90002-2]
- Pang, L.Y., Dai, G., Cecil, T., Dam, T., Cui, Y., Hu, P., Chen, D., Baik, K., Peng, D., 2008. Validation of inverse lithography technology (ILT) and its adaptive SRAF at advanced technology nodes. *SPIE*, **6924**:69240T. [doi:10.1117/12.775084]
- Poonawala, A., Milanfar, P., 2007a. Mask design for optical microlithography—an inverse imaging problem. *IEEE Trans. Image Process.*, **16**(3):774-788. [doi:10.1109/TIP.2006.891332]
- Poonawala, A., Milanfar, P., 2007b. A pixel-based regularization approach to inverse lithography. *Microelectron. Eng.*, **84**(12):2837-2852. [doi:10.1016/j.mee.2007.02.005]
- Santosa, F., 1996. A level-set approach for inverse problems involving obstacles. *ESAIM Control Optim. Calcul. Var.*, **1**:17-33. [doi:10.1051/cocv:1996101]
- Schellenberg, F.M., 2004. Resolution enhancement technology: the past, the present, and extensions for the future. *SPIE*, **5377**. [doi:10.1117/12.548923]
- Shen, S.H., Yu, P., Pan, D.Z., 2008. Enhanced DCT2-based inverse mask synthesis with initial SRAF insertion. *SPIE*, **7122**:712241. [doi:10.1117/12.801409]
- Shen, Y.J., Wong, N., Lam, E.Y., 2009. Level-set-based inverse lithography for photomask synthesis. *Opt. Expr.*, **17**(26):23690-23701. [doi:10.1364/OE.17.023690]
- Shen, Y.J., Wong, N., Lam, E.Y., 2010. Aberration-aware robust mask design with level-set-based inverse lithography. *SPIE*, **7748**:77481U. [doi:10.1117/12.863973]
- Wong, A.K.K., 2001. Resolution Enhancement Techniques in Optical Lithography. SPIE Press, Bellingham, Washington, USA, p.28. [doi:10.1117/3.401208]
- Wong, B.P., Zach, F., Moroz, V., Mittal, A., Starr, G.W., Kahng, A., 2009. Nano-CMOS Design for Manufacturability Robust Circuit and Physical Design for Sub-65nm Technology Nodes. John Wiley & Sons, Inc., Hoboken, New Jersey, p.27.
- Yu, J.C., Yu, P.C., 2010. Impacts of cost functions on inverse lithography patterning. *Opt. Expr.*, **18**(22):23331-23342. [doi:10.1364/OE.18.023331]
- Yu, P., Pan, D.Z., 2007. TIP-OPC: a New Topological Invariant Paradigm for Pixel Based Optical Proximity Correction. Proc. IEEE/ACM Int. Conf. on Computer-Aided Design, p.847-853. [doi:10.1109/ICCAD.2007.4397370]