An Efficient Mask Optimization Method based on Homotopy Continuation Technique

Frank Liu[†]
IBM Austin Research Lab

Xiaokang (Sean) Shi[‡]
University of Texas at Austin

***Abstract*—In sub-wavelength lithography, traditional resolution enhancement techniques (e.g., OPC) cannot guarantee the optimality of the mask. In this paper, we present a novel inverse lithography method to solve the mask optimization problem. Recognizing that when formulated on a pixel-by-pixel basis with partially coherent optical models, the problem is a large-scale nonlinear optimization problem, we cast the optimization flow into a homotopy framework and apply an efficient numerical continuation technique. Compared to earlier pixel-based inverse lithography methods, our homotopy approach is not only more efficient, but also capable of naturally addressing the mask manufactureability problem. Experiment results in a state-of-the-art lithography environment show that our method generates high fidelity wafer images, and is $100\times$ faster than previously reported inverse lithography method.**

I. Introduction

Lithography is a crucial patterning step in semiconductor manufacturing. The purpose of optical lithography is to use electromagnetic waves (light beams) to transfer patterns on the mask to the photo-resist layer on the wafer surface without making physical contact. The patterns can then be transferred to other materials by the subsequent manufacturing steps.

The resolution of the photo-lithography system,R, can be described by the well-known Rayleigh's Equation:

$$R = k_1 \frac{\lambda}{NA} \tag{1}$$

in which λ is the wavelength of the light source, *NA* is the numerical aperture and k_1 is the factor describing the complexity of resolution enhancement techniques. As the VLSI technology pushes further into the nanometer region, the feasible wavelength of the photo-lithographic system remains unchanged at $193nm$. Although over the years it has been repeatedly predicted that EUVL with the wavelength of $13nm$ would replace traditional optical lithography, large scale commercial availability of EUVL remains uncertain[9]. On the other hand, although the immersion lithography can achieve *NA* of 1.35, it is hard to achieve drastic improvements on *NA* values. Thus it is widely recognized that k_1 remains the most cost effective knob to achieve higher resolution.

Due to the unavoidable diffraction, the optical lithography system is lossy in the sense that only low frequency components of the electromagnetic waves can pass the optical system, resulting in the distortion of the features. As the gap between the required feature size and lithography wavelength gets bigger, the final wafer images are quite different from the patterns on the mask. In the past decade, resolution enhancement techniques (RETs) have become necessary in order to achieve required image fidelity. One well-known RET is the optical proximity correction (OPC), in which the mask patterns are intentionally "pre-distorted" to compensate for the loss so that the desired image can be formed. Although optical system models are used in the OPC process (e.g., "model-based" OPC), the traditional OPC processes start the localized optimization from drawn shapes. Thus the optimality of the final solutions is questionable. To overcome this, a lot of heuristics have to be applied. It is not uncommon to see tens of thousands lines of code with numerous empirical parameters in a RET recipe. It's quite doubtful whether such a fragile collection of heuristics can be applied for the future technology node.

The particular mask design problem can be addressed from a different perspective. Instead of locally perturbing the mask pattern to compensate for the loss, we can treat the mask pattern as the input to the band-limited optical system, and the final wafer image as the output. The task of mask optimization then becomes how to "design" a mask so that the desired output can be achieved. This concept is referred as an "image design" problem[18]. However, the problem itself is often ill-posed and the search space is prohibitively large. For example, if integer programming approach is used, the problem size can be easily over 10^6, which renders many brute-force optimization method ineffective. There were some early attempts in the 1990s to find tractable solutions by using methods such as mixed integer programming[19], simulated annealing[7] and random pixel flipping[10]. However, these methods only generated limited academic success.

In recent years, the growing challenge facing sub-wavelength lithography and the increasing complexity of traditional RET routines have made this idea of inverse lithography technique (ILT) more attractive. Representatives of proposed methods include those based on level-set method[1], diffraction orders in frequency domain[16], as well as pixel-based ILT method[5]. A gradient search method for pixel-based ILT was proposed in [13] to overcome the excessive computational cost. However, only coherent and complete incoherent optical models are used. Although it was mentioned in [13] that the method could be extended to partially coherent optical models, no results were presented. The extension to partially coherent model increases the complexity of problem from linear to quadratic[5].

Yet another technical challenge for pixel-based ILT is the manufactureability of the mask. The resulting mask from ILT can be highly fragmented which is difficult to manufacture. Increasing the grid size can alleviate the problem at the cost of image fidelity. The issue was discussed in [11] but no details were given. In [13], a "regularization" approach was applied by adding a "total variation" penalty term (which is essentially the first-order spatial difference of the mask) in the cost function. However, although pertinent in image processing, regularization techniques are quite nebulous in ILT because we actually do not know how the mask should be look like a priori.

In this paper, we present a novel method for pixel-based inverse lithography. Recognizing when partial coherent optical models are used, the mask optimization problem is inherently a nonlinear optimization problem, we cast the optimization flow in a homotopy

[†]Corresponding author. Address: frankliu@us.ibm.com.
[‡]X. Shi contributed to this research project while he was an intern at IBM.

978-3-9810801-7-9/DATE11/©2011 EDAA

framework. The homotopy continuation approach not only drastically speeds up the optimization runtime, but also forces the mask pixels to coalesce into clusters, hence greatly improves the mask manufactureability. The experimental results show that our method not only generates manufactureable masks with excellent image fidelity, but is also $100\times$ faster than previously reported ILT methods.

For the remaining part of this paper, we present the background of optical lithography modeling in Section II. We present the details of our continuation-based inverse lithography method in Section III. The experimental results are presented in Section V, followed by the conclusions in Section VI.

II. Background

A simplified schematic view of optical lithography is shown in Fig. 1. In the optical system ($\mathbf{\Phi(\cdot)}$), the light source is projected through the mask ($\mathbf{M}$) to create a spatially distributed light intensity ($\mathbf{I}$) on the wafer surface. After the chemical reactions of the photo-resist ($\mathbf{R}(\cdot)$), the final wafer image ($\mathbf{Z}$) is formed. More details can be found in references such as [8].

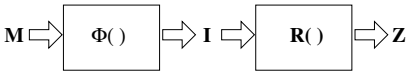

Fig. 1. Schematic view of an optical lithography system.

A. Optical Model

The optical lithography system has been well studied in the past decades. Usually the relationship among light intensity, mask and the optical system is described by the Hopkins model. The light intensity at the wafer surface in frequency domain can be calculated as[12]:

$$I(\omega_x,\omega_y) = \int_{\mathbb{R}^4} M(\xi_1,\xi_2)J_0(\xi_1,\xi_2,\eta_1,\eta_2)M^*(\eta_1,\eta_2) \cdot K(x,y,\xi_1,\xi_2)K^*(x,y,\eta_1,\eta_2)d\xi_1 d\xi_2 d\eta_1 d\eta_2 \quad (2)$$

where $K(\cdot)$ is the point-spread function of the optical system; $J_0(\cdot)$ describes the coherence properties of the light source, or illumination; and $M(\cdot)$ describes the mask pattern. Both $K(\cdot)$ and $J_0(\cdot)$ are translation invariant and furthermore $J_0(\cdot)$ is Hermitian. Thus the Hopkins model in Eqn. (2) can be re-written in vector form as:

$$I(\boldsymbol{\omega}) = \int_{\mathbb{R}^4} \mathbf{M}(\boldsymbol{\xi})J_0(\boldsymbol{\xi}-\boldsymbol{\eta})\mathbf{M}^*(\boldsymbol{\eta})K(\mathbf{x}-\boldsymbol{\xi})K^*(\mathbf{x}-\boldsymbol{\eta})d\boldsymbol{\xi}d\boldsymbol{\eta} \quad (3)$$

the operation is essentially a four-dimension convolution.

The Hopkins model can be drastically simplified in two special illumination scenarios. If the illumination is completely coherent, $J_0(\mathbf{x}) = \delta(\mathbf{x})$, then the intensity calculation can be simplified as:

$$\mathbf{I} = |\mathbf{M} \otimes \mathbf{K}|^2 \quad (4)$$

where $\mathbf{M}$ is the vector representation of the mask, $\mathbf{K}$ is the coherent optical model, and the operator $\otimes$ denotes two-dimensional convolution. Another special case is when the illumination is completely incoherent, then the Hopkins model can be simplified as:

$$\mathbf{I} = |\mathbf{M}|^2 \otimes |\mathbf{K}|^2 \quad (5)$$

For a realistic optical lithography system, the partially coherent model shown in Eqn. (3) has to be used. To reduce the runtime complexity, SOCS (Sum-Of-Coherent-Systems) approximation [3][12] is often used so that the intensity can be calculate as multiple two-dimensional convolutions:

$$\mathbf{I} \approx \sum_{k=1}^{m} \sigma_k |\phi_k \otimes \mathbf{M}|^2 \quad (6)$$

in which σ_k represents the eigenvalues of the optical kernels. Practically almost all contemporary lithography simulators use SOCS approximation.

B. Photo-resist Model

Photo-resist models are necessary to quantitatively describe the chemical reactions on the wafer surface [8]. Typical examples are the constant photo-resist threshold models (CTR) and variable photo-resist threshold models (VTR)[14]. In a CTR model, a simple step-like function is applied to the light intensity on the wafer surface:

$$z(I) = \begin{cases} 1 & \text{if } I \geq t_r \\ 0 & \text{if } I < t_r \end{cases} \quad (7)$$

where t_r is the CTR threshold. The CTR model captures most of the photo-resist behavior and is widely used in traditional RET and inverse lithography methods. In this study, we use the CTR model.

III. Homotopy-Based Inverse Lithography

There are several ways to formulate the pixel-based ILT problem[5]. In this study, we formulate it as the 2-norm between the given target and the wafer image generated by the mask. We further assume a traditional binary mask is used. After spatial discretization, the quality of the wafer image can be described by the following cost function[19]:

$$F(\mathbf{M}) = \frac{1}{N^2}\sum_{j=1}^{N^2}(\hat{z}_j - z_j)^2 \quad (8)$$

where N denotes the number of sampling points in each direction; $\hat{z}$ represents the given image target, and z_j represents the wafer image generated by the mask. The cost function can be easily extended to include a weight function w_j at each location to reflect the importance of image quality for selected regions. Hence the task of mask optimization in ILT is to find the optimal mask $\mathbf{M}$ so that the cost function $F(\mathbf{M})$ is minimized, with the wafer image calculated by the photo-resist model shown in Eqn. (7) and the light intensity by the partially coherent optical model in Eqn. (6). The formal problem definition can be written as:

Problem Definition:
Given the optical model (σ_k, ϕ_k), photo-resist model $z(\cdot)$ and wafer image target $\hat{\mathbf{z}}$, compute the optimal mask $\mathbf{M}$ such that $m_j \in \{0,1\}$, and the cost function $F(\mathbf{M})$ is minimized.

A. Transformation to Continuous Optimization

Since both the mask and wafer image have the range on the discrete set of $\{0,1\}$, the nonlinear mask optimization problem can be solved directly in the discrete domain by using integer programming methods. However, the sheer size of the problem is quite daunting for many IP methods. As indicated in [5], the problems with the size of 300×300 are "*huge*" and are "*unrealistic*" to be solved by today's integer programming methods.

To alleviate the computational issue, the authors of [13] proposed to use transformations to convert the discrete problem to continuous domain, so that more computationally efficient optimization methods can be applied. The first transformation is a sigmoid function to approximate the photo-resist model in Eqn. (7):

$$S_1(I;\gamma,t_r) = \frac{1}{1+e^{-\gamma(I-t_r)}} \quad (9)$$

When γ is sufficiently large, the behavior of this model asymptotically approximates the CTR model in Eqn. (7). This function

has continuous first-order derivative, which is required for many continuous optimization method to work.

To remove the constraint of the binary mask , the authors in [13] proposed to use the transformation $m_j = (1 + cos(\theta_j))/2$. The domain of this function is $\mathbb{R}$ while its range is $[0, 1]$. In other words, instead of optimize on m_j, we can instead optimize on a "fictitious mask", θ_j. Since the domain of the problem is now $\mathbb{R}$, we can apply unconstrained optimization methods, which are usually more efficient than constrained methods. However, no justification on the choice of function $(1 + cos(\theta))/2$ was given in [13]. For those pixels whose optimized values fall between 0 and 1, the thresholding method has to be used to convert the range to $\{0, 1\}$. This practice can affect the image quality and a correction term is needed. Further the authors of [13] only applied the method to coherent or incoherent optical models.

B. Homotopy Continuation Method

Recognizing that the mask optimization is a large-scale nonlinear optimization problem, we propose to cast the problem into a homotopy continuation framework[2][15]. Homotopy method is a well-proven approach to solve difficult nonlinear optimization problems. The basic idea is to argument the **hard** nonlinear problem $G(x)$ with an easier problem $F(x)$ to form a homotopy

$$H(x, \alpha) : \mathbb{R}^N \times \mathbb{R} \rightarrow \mathbb{R}^N \tag{10}$$

such that:

$$H(x, 1) = G(x), \qquad H(x, 0) = F(x) \tag{11}$$

$F(x)$ has the same dimension and is easier to solve (e.g., a linear system). We start with a small homotopy parameter α, so that we solve a easy system. By dynamically adjusting the value of α towards 1, we are forcing the augmented problem morph from easy to heard. Once α reaches 1, we have essentially solved the original hard system $G(x)$. The homotopy methods have been successfully used to solve many large scale nonlinear systems. For instance, it was successfully used to solved the DC problem for large digital circuits with almost 10^4 MOSFETS[17].

Our goal is to argument the ILT problem so that we can continuously "morph" the problem from easy, but far from the exact, to the exact, albeit hard. To achieve this goal, we need to control the "stiffness" of the system. Our approach is to find a "tunable" transformation between the fictitious mask and the real binary mask. There are many good candidates, including one which is right under our nose:

$$m_j = S_2(\theta_j; \alpha) = \frac{1}{1 + e^{-\alpha\theta_j}} \tag{12}$$

When α is small, the shallow slope of the gradient makes it easier for the optimization routine to find a good solution. When α is sufficiently large, the function practically becomes a Heaviside function, which produces conventional binary masks. A family of the function in Eqn. (12) with different values of α, from 0.4 to 8, is plotted in Fig. 2. Using sigmoid function for mask transformation was also independently discovered in [20]. However, [20] didn't see the connection to homotopy continuation and the mask manufacture-ability enhancement, which will be discussed in the next subsection.

After casting the problem into a homotopy framework, our next task is find a numerical method so that the optimal solutions can be quickly achieved. There are plethora of numerical methods on how to find a continuation path[2][15]. In this study we use the predictor-corrector path following method:

$$\alpha_{p+1} = \alpha_p + \Delta\alpha_p \tag{13}$$

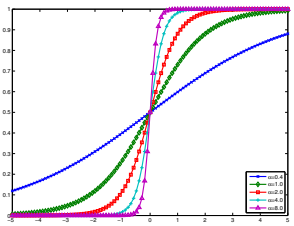

Fig. 2. Mask transformation function with different α values.

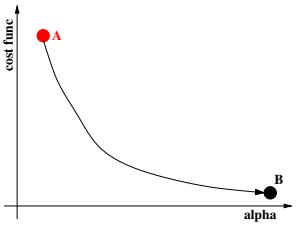

Fig. 3. Illustration of a possible continuation path.

The step size $\Delta\alpha_p$ has to be dynamically adjusted. The details will be described in Subsection III-E.

The inner loop of the homotopy method is a nonlinear optimization problem. After the transformation, our problem is essentially an unconstrained nonlinear optimization problem. Varieties of unconstrained nonlinear optimization methods can be used. We use the steepest-descent method in this study.

$$\theta_{p+1} = \theta_p - h_p \cdot \frac{\partial F}{\partial \theta_p} \tag{14}$$

Here h_p is the step length for each iteration.

Another mechanism we can exploit in the homotopy framework is the number of optical kernels to include. Recall that the kernel eigenvalues σ_k of the SOCS approximation in Eqn. (6) decays rapidly, which means the contributions of higher-order kernels are smaller than the dominant low-order kernels. Therefore, the number of kernels to be included can be treated as another homotopy parameter. We can start with a few dominant kernels and add more kernels on the continuation path.

C. Coalescing Effect of Homotopy Transformation

Yet another benefit of the tunable homotopy transformation is that we can control the "clustering" of the mask. On the left of Fig. 4, the Jacobians $\partial F/\partial\theta_j$ at different pixel locations for a square target for $\alpha = 0.4$ is shown. We further plot the Jacobians in the x-direction with α values of 0.4 and 9.0 on the right of Fig. 4, overlaid with the target. Note that when α is small, the Jacobians spread smoothly across the square target boundary, hence they drive the mask pixels towards positive in a smooth and continuous way. However, when α is large, the sharp transition at the target boundary prevents further movement of the unwanted fragmented pixels. By dynamically tuning homotopy parameter α from small to large, we are forcing the mask pixels to coalesce into large clusters.

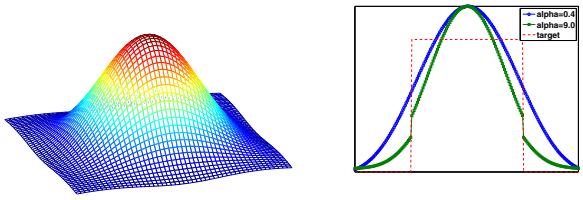

Fig. 4. LEFT: Jacobians of a square target. RIGHT: Jacobians in x-direction of two different alpha values, overlaid with the target.

D. Analytical Calculation of the Gradient

With the introduction of two transformations, the gradient of the mask with respect to the cost function can be analytically calculated.

Use the continuous form of photo-resist function in Eqn. (9), the cost function in Eqn. (8) can be written as:

$$F = \frac{1}{N^2} \sum_{j=1}^{N \times N} (\hat{z}_j - S_1(I_j))^2 \tag{15}$$

We expand the partially coherent optical model in Eqn. (6) into a matrix-vector form below:

$$I_j = \sum_{k=1}^{m} \sigma_k \left| \sum_{i=1}^{N \times N} \phi_{jki} \cdot m_i \right|^2 \tag{16}$$

where m_i represents the mask at location i. We introduce the fictitious mask $\mathbf{\Theta}$ and use the tunable homotopy transformation in Eqn. (12), the light intensity then becomes:

$$I_j = \sum_{k=1}^{m} \sigma_k \left| \sum_{i=1}^{N \times N} \phi_{kji} \cdot S_2(\theta_i; \alpha) \right|^2 \tag{17}$$

The derivative of $F(\mathbf{\Theta})$ with respect to the mask at a location x is:

$$\frac{\partial F}{\partial \theta_x} = -\frac{2}{N^2} \sum_{j=1}^{N \times N} (\hat{z}_j - S_1(I_j)) \cdot S_1^{'}(I_j) \frac{\partial I_j}{\partial \theta_x} \tag{18}$$

Recall for any complex number, the magnitude can be calculated as the product of itself and its complex conjugate:

$$|a|^2 = a \cdot a^* \tag{19}$$

and the differentiation property of convolution:

$$\frac{\partial}{\partial x}(f \otimes g) = \frac{\partial f}{\partial x} \otimes g = f \otimes \frac{\partial g}{\partial x} \tag{20}$$

after a great deal of simplification, we can compute the derivative of the cost function F with respect to the fictitious mask at location j, θ_j, as:

$$\begin{aligned} \tfrac{\partial}{\partial \theta_j} F \quad = \quad & -\tfrac{4}{N^2} S_2^{'}(\theta_j) \cdot \\ & \cdot \textstyle\sum_{k=1}^{m} \sigma_k Real\{\sum_{i=1}^{N^2} \phi_{jki} \cdot (\hat{z}_i - z_i) S_1^{'} f_{ik}^*\} \end{aligned} \tag{21}$$

where f_{ik} is the field at location i from contributions of kernel k:

$$f_{ik} = \sum_{j=1}^{N^2} \phi_{ikj} S_2(\theta_j) \tag{22}$$

Notice that the operation on kernel $\phi(\cdot)$ in Eqn. (21) is yet another convolution. If we store the field from each kernel in Eqn. (22), then the additional computational cost of the gradient is one more convolution per kernel.

E. Details of the Mask Optimization Flow

Putting all pieces together, our method consists of an outer homotopy loop and an inner steepest descent loop. Theoretically we should proceed with the outer loop until the inner loop has converged. However, since in each homotopy iteration we are solving an approximation of the original problem, there is actually no need to reach complete convergence. The extra computation effort will be discarded anyway. Therefore, we intermingle the homotopy loop and the steepest-descent loop.

The homotopy step size in Eqn. (13) can be calculated in different ways[2]. To reduce computational cost, we use a simple scheme by constructing a first-order model between the reduction rate of the cost function and the homotopy step size. If the linear relationship holds, we increase the homotopy step size. If not, we reduce the step size. When the cost functions between two subsequent steps are increasing in stead of decreasing, it gives us an indication that the search has just passed a ridge separating two local optima. In this case, we keep the homotopy parameter α at the current value so that the inner nonlinear search loop has enough time to reach the next downhill slope. The overall flow is summarized in Algorithm 1.

Algorithm 1 Homotopy Continuation Inverse Lithography

Require: Optical model (σ_k, ϕ_k), photo-resist model $z(t_r)$
Require: Wafer image target $\hat{z}$
Require: Parameters F_{min}, α_{max}, β_1, β_2 and $\Delta\alpha_{max}$
Initialize θ_j, α and $\Delta\alpha$
Select k kernels to use
loop
  while ($F > F_{min}$) and ($\alpha < \alpha_{max}$) **do**
    Calculate intensity with fictitious mask θ_j using k kernels
    Calculate cost function and gradient $F(\theta_j, \alpha)$, $\frac{\partial F}{\partial \theta_j}$
    $\theta_j \leftarrow \theta_j - h \cdot \frac{\partial F}{\partial \theta_j}$
    $\alpha \leftarrow \alpha + \Delta\alpha$
    if $F \geq F_{previous}$ **then**
      $\Delta\alpha \leftarrow 0$
    else
      Fit ΔF vs $\Delta\alpha$
      if (linear projection holds) **then**
        $\Delta\alpha \leftarrow \beta_1 \times \Delta\alpha$
        if $\Delta\alpha \geq \Delta\alpha_{max}$ **then**
          $\Delta\alpha \leftarrow \Delta\alpha_{max}$
        end if
      else
        $\Delta\alpha \leftarrow \beta_2 \times \Delta\alpha$
      end if
    end if
  end while
  $k \leftarrow k + \Delta k$
  if ($k \geq k_{max}$) **then**
    break
  end if
end loop

F. Complexity Analysis

For the problem with over 1000×1000 pixels in each frame, the runtime is dominated by the computation of the gradient, which requires two convolutions per kernel (one for the intensity itself, one for the sensitivities). The most straight-forward method to implement 2D convolution is through FFT. Note that even if the Fourier transformations of the kernels are pre-calculated, we still need two FFT operations per convolution. Since the FFT has the complexity of $O(nlog(n))$, the complexity of gradient computation is $O(N^2 log(N^2))$ where N is the number of pixels in each direction. For problems where the changes are localized, a local update scheme can also be used[5].

IV. Mask Manufacturability Enhancement

As we have discussed earlier, the small homotopy parameter introduces the coalescing force to the mask pixels. As the homotopy parameter is dynamically adjusted, the mask pixels are gradually "frozen" to relatively large clusters. This is a much natural and effective method than adding the spatial difference as the penalty term.

After the optimization, we apply a post-processing step to further enhance the manufactureability of the mask by projecting the binary

mask to a Manhattan grid while maintaining the image fidelity. The details of the method were available in [6]. The method uses the dynamic programming approach to compute a globally optimal solution for each feature. Given the efficiency of dynamic programming method, the extra CPU time required is negligible in the overall ILT flow.

V. Experimental Results

Our homotopy inverse lithography method has been implemented in C, using FFTW[4] as the FFT engine for convolution. The optical model and photo-resist models are from a $32nm$ industrial lithography environment. The number of pixels in each frame is 1600×1600. Therefore, for each frame, we need to calculate the binary mask values at 2.5 million locations. The hardware for the benchmarking is a desktop computer with two 3.0-GHz dual-core Intel 5160 Xeon microprocessors and 16 GB of physical memory.

A. Example in 32nm

The first example is a lower level metal routing on a dense regular structure. The original target, the mask generated by our algorithm, the corresponding light intensity on the wafer surface, as well as the wafer image are tabulated from left to right in Fig. 5. Also note the jogs generated by our mask optimization method. They compensate the loss of fidelity due to the diffraction effects. It took our ILT algorithm 144 iterations to achieve convergence with a runtime of 147 seconds, at an average of 1 seconds per iteration. The continuation path is shown in Fig. 7. Note that the cost function is normalized as the percentage of the total number of pixels in the frame and plotted in the logarithmic scale. The initial cost function of 40 means nearly one in very two pixels in the frame are different from the target. However, our algorithm quickly find the right search direction in the first 10 steps.

B. Quantitative Comparison in 22nm

To further demonstrate the power of our mask optimization method, we apply our ILT method to N+1 generation technology. All examples in this subsections are taken from designs in $22nm$ technology. The fidelity of the images from our masks (essentially $32nm$ masks) and from standard $22nm$ masks are compared. Note that the standard $22nm$ masks are significantly more expensive than $32nm$ masks. Again the fidelity is normalized by the total pixel numbers within the frame.

Nine frames are used as the benchmarks and the results are tabulated in Table. I. We also list the number of iteration our optimization method used to achieve convergence. As indicated in the table, except in one case, our masks have consistently better fidelity than the standard $22nm$ mask, with the average number of iterations of 120.

Frame	Error std 22nm mask	Error ILT mask	# iterations
8ts2mar	2.016%	1.943%	119
8ts3mar	1.973%	1.828%	117
l1crp	2.318%	2.508%	179
crand2	1.675%	1.238%	97
ctplcb	2.025%	1.744%	146
base5lc	1.923%	1.651%	117
clkb	2.528%	1.458%	122
dibcl	1.900%	1.426%	123
p4dly	2.370%	1.858%	133

TABLE I
Quantitative comparison of the wafer image fidelity.

We select frame "8st3mar" and show the wafer images from both standard $22nm$ mask and from our ILT mask in Fig. 6. The mixture of both horizontal and vertical features as well as their small geometrical scale make this particular frame challenging. Note the expanded line ends in the ILT mask, which resembles the "hammer head" generated by OPC procedures. Also note the isolated small shapes which function as the sub-resolution assist features (SRAF). The fact that our optimization algorithm can automatically compute the locations and sizes of those assist features demonstrates that our algorithm indeed has converged to an optimal solution. We also show the details of the ILT mask in Fig. 6 which does not contain the fragmented features as in other ILT methods.

C. Runtime Comparison

It is difficult to conduct a direct runtime comparison with the methods proposed in [13] and [5] due to the unavailability of the comparison code and optical models. As an indirect comparison, the method in [13] routinely took 200 iterations to achieve convergence, even for coherent optical models, while our homotopy ILT method took roughly half. Another indirect comparison is that the method in [13] achieved convergence for a 16×16 frame in 5 seconds on a 1.4-GHz Pentium-M computer. As the best-case scenario, we assume their algorithm has linear runtime complexity with respect to the number of pixels within a frame (even for partially coherent optical models). Following the linear extrapolation, the runtime on a 1600×1600 frame would have been 50,000 seconds. Assuming the improvement of the hardware can achieve $4\times$ performance gain, the method in [13] would take 12,500 seconds on the same hardware platform. On the other hand, the average runtime of our ILT method is 120 seconds, which is equivalent to a $100\times$ speed-up.

D. Dense Regular Pattern in 22nm

Our last example is a frame with dense regular patterns. The small features further stress the capability of the resolution enhancement method. The mask generated by our ILT method and the wafer image are shown in Fig. 8. It took our homotopy ILT method 112 iterations to achieve convergence. Notice the counter-intuitive features in the mask, which are hard for traditional OPC routines to produce. However, our ILT method correctly identified those sub-resolution features.

VI. Conclusion

In this paper, we present an efficient inverse lithography optimization method for deep sub-wavelength lithography. The method utilizes homotopy continuation method to achieve fast convergence for the large-scale nonlinear optimization problem stemmed from pixel-based inverse lithography problem. The continuous continuation transformation also provides a natural way to enhance the mask manufactureability. The experimental results in a state-of-the-art lithography environment and industrial designs demonstrate that the method is highly effective. For future work, we plan to incorporate other mask optimization aspects into our optimization flow.

References

[1] D. S. Abrams and L. Pang. Fast inverse lithography technology. In *Proceedings of SPIE*, volume 6154, page 61541J, 2006.

[2] E. L. Allgower and K. Georg. *Numerical path following*. John Wiley & Sons, New York, NY, 1994.

[3] N. Cobb. *Fast optical and process proximity correction algorithms for integrated circuit manufacturing*. PhD thesis, University of California at Berkeley, 1998.

[4] M. Frigo and S. G. Johnson. The design and implementation of FFTW3. *Proceedings of IEEE*, 93(2):216–231, 2005.

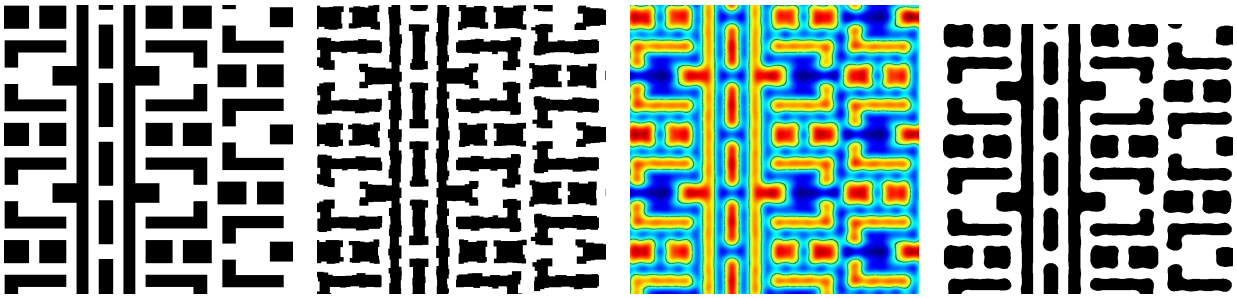

Fig. 5. Example of a dense regular design in $32nm$. From left to right: target, mask, wafer intensity and corresponding wafer image

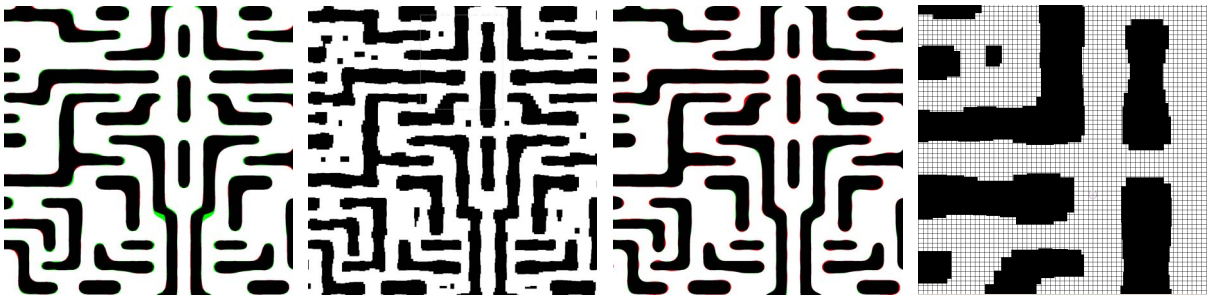

Fig. 6. Frame "8ts3". From left to right: wafer image from standard 22nm mask (Overlaid with the target. Red and green area indicates the difference between the target and wafer image); ILT mask; image from ILT mask, also overlaid with target; details of the ILT mask.

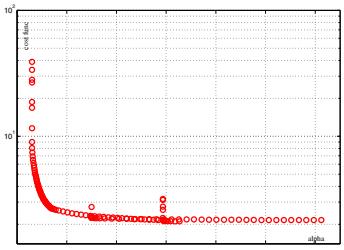

Fig. 7. Homotopy convergence path of the first example. X-axis is the homotopy parameter; Y-axis is the cost function.

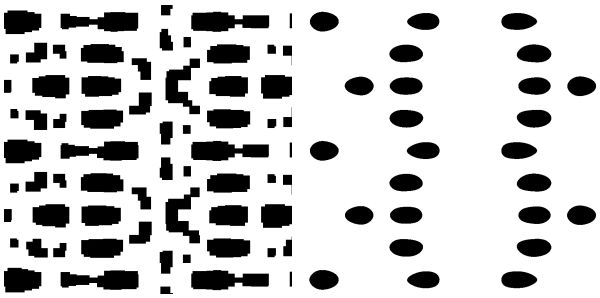

Fig. 8. A dense regular pattern in $22nm$. LEFT: mask generated by our ILT method. RIGHT: wafer image.

[5] Y. Granik. Fast pixel-based mask optimization for inverse lithography. *J. Microlitho., Microfab., Microsyst.*, 5(4), Oct-Dec 2006.

[6] F. Liu et al. Fracturing continuous photolithography masks. U.S. Patent Pending, 2010.

[7] Y. Liu and A. Zakhor. Binary and phase shifting mask design for optical lithography. *IEEE Transactions on Semiconductor Manufacturing*, 5(2):138–151, 1992.

[8] C. A. Mack. *Fundamental Principles of Optical Lithography: The Science of Microfabrication*. John Wiley & Sons, W. Sussex, England, 2007.

[9] C. A. Mack. Seeing double. *IEEE Spectrum*, 45(11):46–51, November 2008.

[10] Y. Oh, J. C. Lee, and S. Lim. Resolution enhancement through optical proximity correction and stepper parameter optimization for 0.12-μm mask pattern. In *Proc. SPIE Optical Microlithography*, volume 3679, pages 607–613, 1999.

[11] L. Pang, Y. Liu, and D. Abrams. Inverse Lithography Technique, What is the impact to Phtomask Industry. In *Proceedings of SPIE*, volume 6283, page 62830X, 2006.

[12] Y. C. Pati and T. Kailath. Phase-shifting masks for microlithography: automated design and mask requirements. *Journal of the Optical Society of America A*, 11(9):2438–2452, 1994.

[13] A. Poonawala and P. Milanfar. Mask design for optical microlithography-an inverse imaging problem. *IEEE Transactions on Image Processing*, 16(3):774–788, 2007.

[14] J. Randal, K. Ronse, T. Marschner, M. Goethais, and M. Ercken. Variable-threshold resist models for lithography simulation. In *Proc. of SPIE 3679*, pages 176–182, July 1999.

[15] S. Richter and R. DeCarlo. Continuation methods: Theory and applications. *IEEE Transactions on Automatic Control*, 28(6):660–665, 1983.

[16] A. E. Rosenbluth, S. Bukofsky, C. Fonseca, M. Hibs, K. Lai, A. Molless, R. N. Singh, and A. K. K. Wong. Optimum mask and source patterns to print a given shape. *Journal of Microlithography, Microfabrication and Microsystem*, pages 13–30, April 2002.

[17] J. Roychowdhury and R. Melville. Delivering global DC convergence for large mixed-signal circuits via homotopy/continuation methods. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(1):66–78, 2006.

[18] S. Sayegh, B. Saleh, and K. Nashold. Image design: generation of a prescribed image through a diffraction limited system with high-constrast recording. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 33(2):460–465, 1985.

[19] S. Sherif, B. Saleh, and R. D. Leone. Binary image synthesis using mixed linear integer programming. *IEEE Transactions on Image Processing*, 4(9):1252–1257, September 1995.

[20] J. Zhang et al. A robust pixel-based RET optimization algorithm independent of initial conditions. In *Proceedings of the Asia/South Pacific Design Automation Conference*, 2010.