JM3 Journal of Micro/Nanopatterning, Materials, and Metrology

RESEARCH PAPER

Ultra-fast aerial image simulation algorithm using wavelength scaling and fast Fourier transformation to speed up calculation by more than three orders of magnitude

Tsai-Sheng Gau,[a],* Po-Hsiung Chen,[a] Burn J. Lin,[a] Fu-Hsiang Ko,[b] Chun-Kung Chen,[c] and Anthony Yen[d]

[a]National Tsinghua University, College of Semiconductor Research, Hsinchu, Taiwan
[b]National Yang Ming Chiao Tung University, Department of Material Science and Engineering, Hsinchu, Taiwan
[c]ASML Technology Development Center, Hsinchu, Taiwan
[d]ASML Technology Development Center, San Jose, California, United States

ABSTRACT. An ultra-fast image simulation algorithm is proposed. The new algorithm uses full fast-Fourier-transform (FFT) to calculate the aerial image intensity. The wavelength, 193 nm, was scaled to a number of powers of 2, through scaling the mask with a scaling factor derived from the discrete Fourier transform (FT) format. The mask can then be transformed to the diffraction spectrum in terms of spatial frequency using the FFT algorithm. Similarly, this mask diffraction spectrum can be inverse transformed to the aerial-image by using the inverse-FFT algorithm. The image is finally scaled back to the original image amplitude of the original wavelength and squared to the image intensity. Comparing to the original FT, there is a 4000× to 5000× computation speed improvement with only about 3% intensity deviation. This algorithm provides an efficient engine for lithography optimization.

[DOI: 10.1117/1.JMM.22.2.023201]

Keywords: fast Fourier transform; lithography; lithography simulation; resolution enhancement technology

Paper 23017G received Mar. 18, 2023; revised May 24, 2023; accepted May 30, 2023; published Jun. 14, 2023.

1 Introduction

Simulations are intensively used in semiconductor lithography processes, no matter during process development or mass production of IC chips. The computational lithography, generally consisting of optical proximity correction (OPC),[1–7] source mask optimization (SMO),[8–14] inverse lithography (ILT),[15–17] and other simulation packages,[18–20] is widely and accurately applied during different phases of process development. At the path-finding stage, new types of phase-shifting mask, resist materials, and resolution-enhancement techniques, should be thoroughly studied and simulated to find opportunities for pushing the patterning resolution. After moving to the process development stage, we need to simulate the process windows to establish the design rules for each masking layer. Before that, the optimum numerical aperture (NA) and the partial coherence parameter σ of the exposure tools have to be evaluated. Moreover, for modern lithography, it needs SMO, OPC, pattern splitting (for multiple patterning), and even ILT, if the resolution scaling coefficient k_1 is beyond the control of conventional lithography process.[21,22]

*Address all correspondence to Tsai-Sheng Gau, tsaisheng_kao@mx.nthu.edu.tw

Even at the mass production stage, especially for the foundry with product-layout diversity, to prevent the degradation of wafer yield from patterning hotspots, sophisticated simulations are needed to detune OPC or SMO to remove the hotspots encountered.

There are two performance indicators for the widely used simulations in lithography technology: accuracy and speed. For cutting-edge semiconductor technology, the systematic critical-dimension (CD) error tolerance is less than a few nanometers. For the 12-nm gate length, the CD error specification is just ±1.2 nm, 10% of the patterning CD. Similar but looser CD tolerance are required for the metal and the contact layers. The mask CD must just be located at the center of the wafer random CD-error distribution. A desirable OPC package is used to render the simulated mask CD accuracy by the model and rules derived from all the predictable patterning effects, such as optical, resist, etching and even chemical–mechanical polish.[23] In addition, since there are billions of transistors in a single chip, a speedy simulation is necessary. For example, the 10-nm technology node requires the simulation software to complete the simulation and correction work in less than a week. The computer CPU cores involved could be more than several tens of thousands for multiple products handled simultaneously.

The algorithm presented in this paper can speed up the transformation between the physical space and the spatial frequency space by making typical imaging simulations containing the wavelength parameter to fit the fast-Fourier-transform (FFT) scheme. Depending on the number of integrations needed, typical imaging simulations can be sped up by more than three orders of magnitude.

2 Theory

2.1 Basic Simulation

The basic lithographic simulation, no matter the purpose, must include aerial image simulation. For Fourier optics, it is the FT of the amplitude and phase distribution $M(\vec{x})$, through the mask; then, inverse-FT back to the spatial domain after multiplying with a low-pass pupil function, $P(\vec{k})$. The aerial image intensity $I(\vec{x})$ of a point coherent light source.[24–26] is defined as

$$I(\vec{x}) = \left| \int_{-\infty}^{\infty} P(\vec{k}) e^{2\pi i(\vec{k}\cdot\vec{x})} \mathrm{d}\vec{k} \int_{-\infty}^{\infty} M(\vec{x}) e^{-2\pi i(\vec{k}\cdot\vec{x})} \mathrm{d}\vec{x} \right|^2, \tag{1}$$

where $\vec{x}$ is the spatial position, $\vec{k}$ is the wave-vector with amplitude $\frac{1}{\lambda}$, and λ is the wavelength of the imaging light. For partially coherent light with distribution $\sigma(\vec{k}')$, the final aerial image intensity is the summation of the individual intensity produced by each coherent source point $\vec{k}'$ and is governed by Eq. (2), where Hopkin's approximation is used:

$$I(\vec{x}) = \int_{\sigma} \sigma(\vec{k}') \mathrm{d}\vec{k}' \left| \int_{-\infty}^{\infty} P(\vec{k} - \vec{k}') e^{2\pi i(\vec{k}\cdot\vec{x})} \mathrm{d}\vec{k} \int_{-\infty}^{\infty} M(\vec{x}) e^{-2\pi i(\vec{k}\cdot\vec{x})} \mathrm{d}\vec{x} \right|^2 \tag{2}$$

The low-pass-filter pupil function, $P(\vec{k})$, as a function of the illumination distribution $\sigma(\vec{k}')$ can be expressed as the following:

$$P(\vec{k}) = \begin{cases} = 1, & (k_x^2 + k_y^2)^{1/2} < \frac{NA}{\lambda} \\ = 0, & (k_x^2 + k_y^2)^{1/2} > \frac{NA}{\lambda} \end{cases}, \tag{3}$$

where NA, the numerical aperture, is defined as $n \sin \theta$, n is the refractive index of the imaging medium, and $\sin \theta$ is the maximum allowed directional component of $\vec{k}$ to pass through the lens pupil to contribute to the aerial image. On the other hand, the illumination distribution $\sigma(\vec{k}')$ can also be expressed as

$$\sigma(\vec{k}') = \begin{cases} = 1, & (k'^2_x + k'^2_y)^{1/2} < \frac{\sigma}{\lambda} \\ = 0, & (k'^2_x + k'^2_y)^{1/2} > \frac{\sigma}{\lambda} \end{cases}, \tag{4}$$

where σ is the partial coherence parameter of the illumination system of an exposure tool. The illumination distribution in Eq. (4) is a conventional disk illumination. For advanced lithography, the illumination distribution is customized for lithography performance and can be an arbitrary distribution within unity, i.e., $\sigma = \frac{n \sin \theta_i}{NA} < 1$. Here, $n \sin \theta_i$ is the illumination NA, and θ_i is the angle between the incident beam to the mask normal. The intensity obtained from Eq. (2) is based on the TE polarization, linear polarization with the direction of the electric field tangential to the wafer surface, and the diffraction amplitudes are scalar summed. For the TM polarization, Eq. (2) needs to be corrected with the vector summation of the individual ampltitude.[19,24] The pupil function $P(\vec{k})$ in Eq. (3) is an ideal pupil function. Equation (5) shows the real pupil function $P'(\vec{k})$, which is the ideal pupil function multiplied by a wavefront error function $e^{i2\pi W}$:[19,24]

$$P'(\vec{k}) = P(\vec{k})e^{i2\pi W}, \tag{5}$$

where

$$W = \sum_{i=0}^{\infty} Z_i F_i, \tag{6}$$

Z_i is the Zernike coefficient and F_i is the Zernike Polynomial. Among those terms, Z_3 is the defocus term, which needs to be scaled along with other terms based on the new algorithm and will be discussed later. The computation of Eq. (2) is the core step of lithography simulation. Regardless of the application, be it for phase-shifting mask, mask three-dimensional effect, aberration, defocus, or even ILT, there is a need for a fast aerial image calculation as a base to build advanced applications for development of the realistic semiconductor technology.

2.2 Numerical Calculation

There are several ways to numerically calculate the result of Eq. (2). The direct calculations consist of the Abbe and the transmission cross coefficient (TCC) approaches.[24,25,27] The former approach traces each illumination point source and incoherently adds up the intensities. The latter approach calculates the transmission cross coefficient (TCC) matrix in advance. The TCC matrix is the integral of the product of the illumination distribution, pupil function, and the complex conjugate of the pupil function, which is shifted by the incident wave vector. The advantage of the TCC approach is that the TCC needs to be calculated only once after fixing the illumination and the pupil function. Therefore, this method is very suitable for varied mask layout during each simulation. However, direct computation is very time consuming, due to the deep loops of FT. Another approach is to diagonize the TCC matrix in the $\vec{k}$ space or the $\vec{x}$ space to get the eigenvalues and eigenfunctions. The final aerial image intensity can be obtained by summing the convolutions of the eigenfunction and mask with multiplying the eigenvalues. This leads to the sum of coherence system (SOCS) method.[25,27] It is most widely adopted in commercial software packages, especially for the OPC software.

2.3 Fourier Transform and Fast Fourier Transform

The FT and inverse FT of M and A are as follows:

$$A(k_x, k_y) = \int_{-\infty}^{\infty} M(x, y)e^{-2\pi i(k_x x + k_y y)} \mathrm{d}x\,\mathrm{d}y \quad M(x, y) = \int_{-\infty}^{\infty} A(k_x, k_y)e^{2\pi i(k_x x + k_y y)} \mathrm{d}k_x\,\mathrm{d}k_y. \tag{7}$$

The FFT is a well-developed methodology for quick evaluation of discrete FT. Since invented[28] in 1965, the scheme has been implemented in many applications and with several representations.[29] The basic and original algorithm uses radix-2. Equation (6) shows the discrete FT pair

$$A(p, q) = \sum_{m=0,n=0}^{N-1} M(m, n)e^{\frac{-2\pi i(pm+qn)}{N}} \quad M(m, n) = \sum_{p=0,q=0}^{N-1} A(p, q)e^{\frac{2\pi i(pm+qn)}{N}}, \tag{8}$$

where $A(p, q)$ and $M(m, n)$ form a discrete FT and inverse-FT pair; p, q, m, n, and N are integers. If N is a number of powers of 2, $N = 2^r$, in Eq. (8), one of the series can be divided into two parts, with even and odd order components. We use the one-dimensional (1-D) case for convenience of illustration:

$$A(p) = \sum_{n=0}^{N-1} M(n)e^{-2\pi ipn/N} = \sum_{n=0}^{\frac{N}{2}-1} M(2n)e^{-2\pi ip(2n)/N} + \sum_{n=0}^{\frac{N}{2}-1} M(2n+1)e^{-2\pi ip(2n+1)/N}$$
$$= \sum_{n=0}^{\frac{N}{2}-1} M(2n)e^{-2\pi ip(2n)/N} + e^{-2\pi ip/N} \sum_{n=0}^{\frac{N}{2}-1} M(2n+1)e^{-2\pi ip(2n)/N} = C_{N/2}^{2np} + e^{-2\pi ip/N} D_{N/2}^{2np}, \tag{9}$$

where $C_{N/2}^{2np}$ and $D_{N/2}^{2np}$ are two sub-series of $A(p)$. Based on Eq. (8), we can derive $A(p+N/2)$ by substituting p with $p+N/2$ and remember that p and $p+N/2$ are both integers:

$$A\left(p+\frac{N}{2}\right) = C_{N/2}^{2np} - e^{-2\pi ip/N} D_{N/2}^{2np}. \tag{10}$$

Equation (8) shows that it can save half of the computation. This is because when $A(p)$ is calculated, $A(p+N/2)$ is known by using Eq. (10). If N is a number of powers of 2, the procedure can be iteratively executed, until only two terms left. If an FT needs $N \times N$ computations, the FFT can reduce the computations to $N \log_2 N$.

To directly calculate the aerial image intensity distribution of Eq. (2), when we set 64 pixels in 1-D, it needs 64^4 computations to calculate the two-dimensional (2-D) mask spectrum plus 64^6 computations for the final aerial image intensity. An extra 64^2 computations is needed for scanning the illumination. It is an enormous task, if no other algorithm is used to boost up the computation. This paper uses the FFT to compute the aerial image, as an in-house development engine for research purpose at the expense of a slight loss of accuracy. When the wavelength is not an integer of the powers of 2, it is scaled to such an integer to enable FFT, then scaled back to the original value after all the operations in FFT and inverse FFT. This procedure induced the aforementioned slight loss of accuracy. For schools or research organizations with limited resource to purchase speedy computation equipment or to access industrial simulation packages, this paper provides a powerful algorithm to convert the traditional FT to FFT to save the enormous computational efforts.

2.4 Fourier Transformation in Summation Form

It is advantageous to convert the FT of the integral form to the summation form. We use the 1-D case for the mask distribution $M(\vec{x})$ and the spatial frequency spectrum $A(\vec{k})$ to demonstrate our methodology:

$$A(\vec{k}) = \int_{-\infty}^{\infty} M(\vec{x})e^{-2\pi i(\vec{k}\cdot\vec{x})}\mathrm{d}\vec{x} \quad M(\vec{x}) = \int_{-\infty}^{\infty} A(\vec{k})e^{2\pi i(\vec{k}\cdot\vec{x})}\mathrm{d}\vec{k}. \tag{11}$$

We convert Eq. (11) into the summation form with discrete sampling

$$A(k) = \sum_{x=-\infty}^{\infty} M(x)e^{\frac{-2\pi i}{\lambda}(k_x x)} \quad M(x) = \sum_{k_x=-\infty}^{\infty} A(k)e^{\frac{2\pi i}{\lambda}(k_x x)}. \tag{12}$$

Equation (6) cannot be directly used to calculate Eq. (12) for the following two reasons. First, the summation limit of Eq. (8) is N, whereas the summation limit of Eq. (12) is infinite. Second, the Fourier pair of Eq. (8) is symmetrical, but those are not symmetrical of Eq. (12). Because the clear-tone masks, transparent patterns and opaque background, are mostly used for EUV lithography and the pupil function to cut off high-order frequencies is multiplied to Eq. (12), the infinity summation limit in Eq. (12) can be replaced by the finite range of interest. On the other hand, the sampling of k_x and x is very asymmetrical. For ArF exposure tools, the exposure wavelength λ is 193 nm. When we set the sampling pixel number to 64 and the pixel size to 25 nm, the sampling range in spatial coordinate x is 1600 nm. In the k space, the k_x, the direction component of wave vector with wavelength λ separated, scan range is normally ± 2 by considering the maximum NA and illumination partial coherence parameter, both are ± 1. But, the scan variables m, n and p, q in Eq. (7) are integers ranging from 0 to $N-1$, which are symmetrical. More over, the challenging part to use FFT to boost the computation speed for this system is that λ is 193, definitely not in powers of 2. However, Eq. (12) can be converted to the format of Eq. (8). Let us use 25 nm for the sampling interval Δx, $\frac{4}{64}$ for Δk_x and 193 nm for λ to substitute into the first part of Eq. (12), we obtain Eq. (13) with pixel number 64:

$$A(r\Delta k_x) = \sum_m M(m\Delta x)e^{-2\pi i\frac{1}{123.52}(rm)}. \tag{13}$$

Now, r and m are both integers ranging from 1 to 64. Equation (13) is the same format as Eq. (8).

Consider two masks, $M(x)$ and $M'(x')$, illuminated with wavelengths λ and N, respectively. If these two systems can produce the same mask diffraction spectrum $A(k)$, what is the relationship between $M(x)$ and $M'(x')$? In the following, we omit the $\pm\infty$ limits in the integral for clarity:

$$A(k) = \int M(x)e^{-2\pi i\frac{1}{\lambda}(k_x x)}\mathrm{d}x, \tag{14a}$$

$$= \int M'(x')e^{-2\pi i\frac{1}{N}(k_x x')}\mathrm{d}x'. \tag{14b}$$

With the inverse Fourier transform (FT), we can get $M'(x')$ with Eq. (14)

$$M'(x') = \int A(k)e^{2\pi i\frac{1}{N}(k_x x')}\mathrm{d}k_x. \tag{15}$$

By substituting $A(k)$ in Eq. (14a) into Eq. (15), we obtain the relationship between $M'(x')$ and $M(x)$:

$$M'(x') = \int A(k)e^{2\pi i\frac{1}{N}(k_x x')}\mathrm{d}k_x \quad = \int\left(\int M(x)e^{-2\pi i\frac{1}{\lambda}(k_x x)}\mathrm{d}x\right)e^{2\pi i\frac{1}{N}(k_x x')}\mathrm{d}k_x. \tag{16}$$

After changing the integral order of dk_x and dx, Eq. (15) becomes

$$M'(x') = \int\left(\int e^{-2\pi i(\frac{x}{\lambda}-\frac{x'}{N})k_x}\mathrm{d}k_x\right)M(x)\mathrm{d}x \quad = \int \delta\left(\frac{x}{\lambda}-\frac{x'}{N}\right)M(x)\mathrm{d}x \quad = CM\left(\frac{\lambda x'}{N}\right), \tag{17}$$

where $\delta(x)$ is the delta function. Equation (15) indicates that the two masks differ by a scaling factor $\frac{\lambda}{N}$ and an intensity proportional constant C.

Foregoing is a basic concept of diffraction physics. Consider the double-slit diffraction consisting of the slit distance d, wavelength λ and the diffraction angle φ as shown in Fig. 1, the diffraction equation is

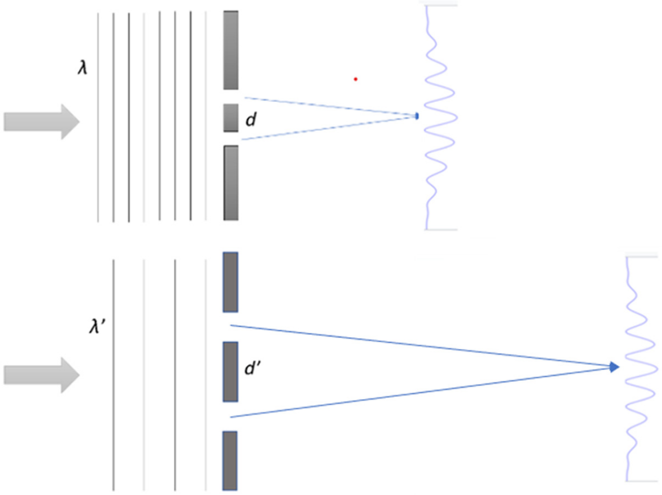

Fig. 1 The double-slit diffraction shows the scaled wavelength and slit geometry to produce the same diffraction spectrum.

$$\frac{2d}{\lambda} = \frac{n}{\sin\,\varphi}. \tag{18}$$

The right side of Eq. (18) shows the mask diffraction spectrum. To maintain the same mask spectrum, the slit distance should be scaled with the wavelength used. Figure 1 shows the scaling of the double slit to produce an identical diffraction spectrum.

In the same way, we now check the image formation after obtaining the mask spectrum. If the same mask spectrum $A(k)$ and pupil function $P(k)$ form the image with different wavelengths and spatial dimensions as shown in Eq. (19), what is the relationship between the image amplitude distribution $E(x)$ and $E'(x')$?

$$E(x) = \int P(k)A(k)e^{2\pi i\frac{1}{\lambda}(k_x x)}\mathrm{d}k_x \quad E'(x') = \int P(k)A(k)e^{2\pi i\frac{1}{N}(k_x x')}\mathrm{d}k_x. \tag{19}$$

Inversing the second part of Eq. (19), we get

$$P(k)A(k) = \int E'(x')e^{-2\pi i\frac{1}{N}(k_x x')}\mathrm{d}x'. \tag{20}$$

Substituting Eq. (20) into the first part of Eq. (19), we get

$$E(x) = \int\left[\int E'(x')e^{-2\pi i\frac{1}{N}(k_x x')}\mathrm{d}x'\right]e^{2\pi i\frac{1}{\lambda}(k_x x)}\mathrm{d}k_x. \tag{21}$$

Following the same procedure in the derivation of Eq. (17), we perform the integration of dk_x before the integration of dx' in Eq. (21), resulting in

$$E(x) = \int\left[\int e^{-2\pi i\left(\frac{x'}{N}-\frac{x}{\lambda}\right)k_x}\mathrm{d}k_x\right]E'(x')\mathrm{d}x' = \int \delta\left(\frac{x'}{N}-\frac{x}{\lambda}\right)E'(x')\mathrm{d}x' = \frac{1}{C}E'\left(\frac{Nx}{\lambda}\right). \tag{22}$$

By squaring E in Eq. (22), the image intensity distribution for wavelength λ is the same as what we get from wavelength N, except for a scaling factor, which is the reciprocal of Eq. (17). The intensity proportional constant C is $(\frac{N}{\lambda})^2$, which will be discussed later. In this paper, the traditional FT can be converted to the discrete FT format in Eq. (12). The discrete FT for mask diffraction spectrum obtained is then calculated by FFT with the mask dimension scaled based on the wavelength ratio from Eq. (17). The mask spectrum obtained with scaled mask is used to calculate the image amplitude by FFT and this amplitude is finally scaled back to the actual image amplitude based on Eq. (22).

3 Simulation

3.1 Simulation Flow

This paper uses *Matlab*®, which is popular and commercially available, as the simulation development platform. The simulation flow aims to approach the conventional FT with the FFT format and algorithm. There are three key steps for the simulation.

First step: Pick N

Set pixel number in 1-D as 64; wavelength, 193 nm; Δx, 25 nm; and Δk_x, $\frac{4}{64}$. The number 4 is the full range of the k value. With normally incident illumination, k ranges from -1 to $+1$. With off-axis illumination in both directions, the full range of k is now from -2 to $+2$. From Eq. (13), $2\pi i\frac{1}{\lambda}(r\Delta x)(m\Delta k_x) = 2\pi i\frac{rm}{123.52} = 2\pi i\frac{rm}{\beta}$, pick the nearest integer of β that is the powers of 2. In this case, set N to 128.

Second step: Scale the mask

From the first step, the scaling factor is $\varepsilon = \frac{N}{\beta} = \frac{128}{123.52}$. The mask $m(\vec{x})$ is scaled up with ε. Because the total pixel number of the mask distribution is 64×64, but N is 128, we need to pad zeros to match the row and column elements of the mask matrix to 128×128. Now, the new mask matrix is ready for FFT.

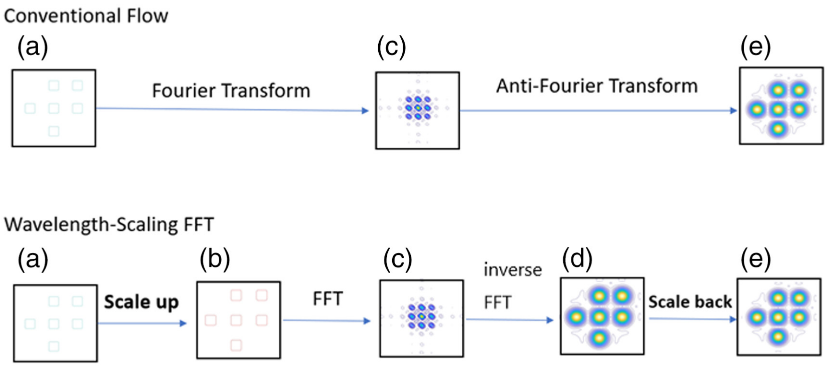

Fig. 2 The simulation flow for conventional FT and wavelength-scaling FFT, with (a) the original mask, (b) the scaled mask, (c) the mask diffraction spectrum, (d) the aerial image before scaling, and (e) the final aerial image after scaling.

Third step: Scale the image

We use the mask spectrum $A(\vec{k})$, from the FFT of the scaled mask $M'(\vec{x'})$, to execute the inverse-FFT for the image amplitude distribution $E'(\vec{x'})$. The image $E'(\vec{x'})$ is scaled with the scaling factor $\frac{1}{\varepsilon}$ to get the final image $E(\vec{x})$.

Figure 2 shows the complete simulation flow, including conventional FT and the wavelength-scaling FFT flows for comparison. The conventional flow follows the FT and inverse-FT based on Eq. (2), and the wavelength-scaling FFT includes the following:

(a) Scale up the original mask $M(\vec{x})$ to the new mask $M'(\vec{x'})$ based on the first and second steps.

(b) Execute FFT to $M'(\vec{x'})$ and obtain the mask spectrum $A(\vec{k})$.

(c) Multiply $A(\vec{k})$ with pupil function $P(\vec{k})$ and execute inverse-FFT to get the image amplitude $E'(\vec{x'})$.

(d) Scale back the intensity $E'(\vec{x'})$ to $E(\vec{x})$ based on the third step.

Although there are two more steps for the new algorithm, the computation speed is much faster than the conventional method.

3.2 Simulation Results

Two cases are used to verify the simulation, a line-and-space pattern and a hole pattern. The critical dimensions and simulation conditions are listed in Table 1.

We use 193-nm wavelength, 64 pixels in 1-D, $\frac{4}{64}$ Δk_x pixel size, 25-nm Δx. The resulting artificial wavelength for the FFT. Figure 3 shows the simulation results, including the original mask, the mask after scaling, the comparison of the 2-D intensity contours and the 1-D intensity cutlines.

The comparison of the simulation results, shown in Fig. 3, consists of the 2-D contours and the 1-D intensity cut lines, evaluated with conventional FT and the wavelength-scaled FFT. They are almost indistinguishable. There is only a slight deviation at the intensity peaks, which is not of the highest interest for lithography. The deviations are mostly from the grid snapping or the

Table 1 The simulation condition of (a) hole case and (b) line/space case.

	(a) Hole case	(b) Line/space case
Layout condition	Hole size 150 nm, hole pitch 425 nm, Fig. 3(a)	Line size 100 nm, line pitch 225 nm, Fig. 3(b)
Optical condition	$\lambda = 193$ nm, NA = 0.5, $\sigma = 0.8$	$\lambda = 193$ nm, NA = 0.7, $\sigma = 0.8$

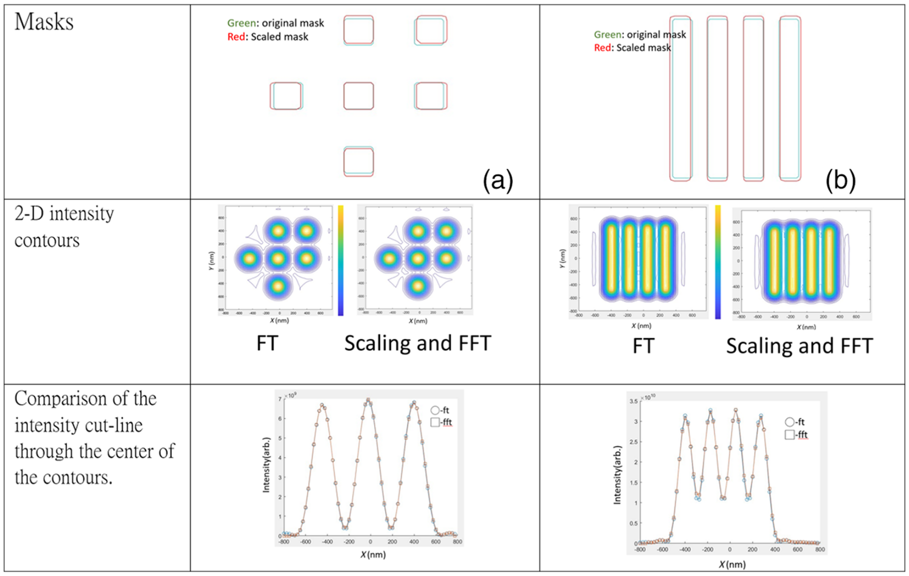

Fig. 3 The simulation results of (a) hole patterns and (b) line and space patterns. FT, conventional Fourier transform; FFT, fast Fourier transform. The intensity cut line is the horizontal line at the center of the 2-D contour.

resolution of the mask governed by the number of pixels. The number of pixels of these simulations, including the mask, is 64×64 and the scaling factor ε is 1.0363. The resolution cannot faithfully reflect the accuracy of the scaling factor during the mask scale up. For example, the mask width of the simulation case in Table 1(b) is 100 nm and the scaling factor ε is 1.03463. The mask width after scaling is about 103.46 nm but the pixel size is 25 nm for the mask. The mask size can only be 100 or 125 nm. To reduce the error, the mask edge was smoothened out with the bilinear interpolation, the transmission becomes not exactly 0 or 1 at the edge of a binary mask. Figure 4 shows the mask intensity distribution after the scaling in contrast to the original binary mask. It is relatively easy to increase the number of pixels to enhance the resolution for the new method. However, it is prohibitive to compare between FT and wavelength-scaled FFT, because the deep loop of conventional FT takes too long to complete, especially when the number of pixels is larger than 100 in each dimension. Table 2 lists the speed and intensity error evaluated with conventional FT and wavelength-scaled FFT for cases (a) and (b). The intensity error here is defined by comparing the integrated intensity under the cut line of the FT and wavelength-FFT, the last part of Fig. 3.

For the real pupil function P', pupil function with aberrated wave front and the Zernike coefficient need to be scaled with the new algorithm. Take Z_3, defocus, as an example[19,24]

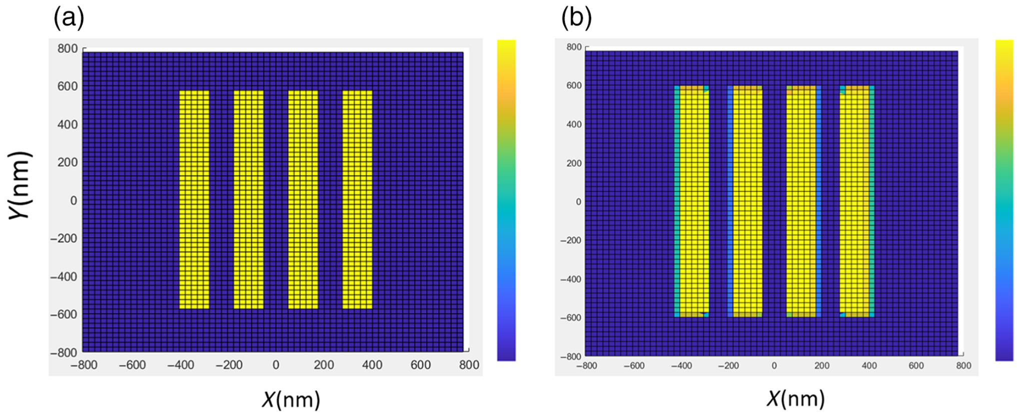

Fig. 4 (a) The original binary mask with transmission 1 and 0. (b) The scaled mask with the transmission approximated by bilinear interpolation.

Table 2 Comparison of simulation results between the conventional FT and the scaled-FFT methods.

	(a) Hole case		(b) Line and space case	
Execution time (s)	Conventional FT	Wavelength-scaling FFT	Conventional FT	Wavelength-scaling FFT
	360	0.077	690	0.16
Intensity error	2.85%		3.2%	

$$Z_3 = \Delta\delta \frac{\mathrm{NA}^2}{4\lambda}, \tag{23}$$

$$F_3 = 2R^2 - 1, \tag{24}$$

$$R = \frac{\lambda}{\mathrm{NA}} \sqrt{k_x^2 + k_y^2}, \tag{25}$$

where $\Delta\delta$ is the focus shift. For the conventional FT, the aerial image intensity can be calculated by replacing the ideal pupil function P in Eq. (2) with the real pupil function P' in Eq. (5) and the Zernike coefficient Z_3, polynomial F_3, respectively, in Eqs. (23) and (24). On the other hand, for the wavelength scaling FFT, Eq. (22) is still valid. But, it needs to scale the Zernike coefficient Z_3 to Z'_3, with $Z'_3 = Z_3/\varepsilon$. This scaling for the wavelength-FFT calculation of the defocus image intensity matches well with the result of FT.

The computation time of the wavelength-scaling FFT method and that of the conventional FT are listed in Table 2. Although there are two extra mask scaling and image amplitude scaling steps, the computation time of the new method way exceeds expectation. In theory, with 64×64 pixels for the 2-D map, mask or intensity, it needs 64^4 computations for the conventional FT. If converted to the wavelength scaled FFT, the computations become $(64 \log_2 64) \times (64 \log_2 64)$. Roughly, it is about 100× runtime saving. There are two major portions for the computation depicted in Fig. 2. Those are step A to C for the first portion and steps C to E for the second portion. For the first portion computation, it is just FT of the mask. It takes 2.7 s for the conventional FT, whereas wavelength FFT only takes 0.025 s, although with one additional step to scale up the mask. It is roughly 108× faster for the new algorithm. For the second portion computation, it needs to scan the whole illumination to sum up the intensities originated from the individual illumination pixel. The inverse FT is executed in this portion. In the FT method, the computation time is just 130×2.7 and 250×2.7 s for the two illumination settings involved in the two cases listed in Table 1. On the other hand, during the illumination scanning, the repeated computation quickly drops from 0.025 s to 5×10^{-4} s for the wavelength-scaling FFT. The overall runtime becomes $130 \times 5 \times 10^{-4}$ and $250 \times 5 \times 10^{-4}$ plus the overhead runtime for the new algorithm. It is possibly due to the new method using very little deep-loop computations with only matrix operations and Matlab® built-in functions, such as *fft2, ifft2, fftshift, ifftshift…* In conclusion, the conversion from FT to FFT saves ~100× runtime. And, because of the coding efficiency of matrix operation, extra 40× to 50× runtime can be saved. It is an extra benefit to convert the coding with matrix operations.

From Table 2, the simulation speed is improved by 4000× to 5000× with the new wavelength-scaled FFT method. It improves the simulation speed at the expense of intensity deviation of the order of 3%. The proportional constants in Eqs. (17) and (22) can be derived by considering the consistency of the flux passing through the mask per unit area. Since the mask was scaled with ε in 1-D, the ratio of the total transparent area in the mask is increased by ε.[2] To keep the mask spectrum consistent after the mask scaling, the mask transmittance after scaling is normalized by $\frac{1}{\varepsilon^2}$. In the same way, we can normalize the final aerial image intensity. Once the normalization constants were fixed, we need to calibrate the results of wavelength-scaled FFT with those of conventional FT only once.

4 Conclusion

This paper reports a new methodology that converts the aerial image calculation from the time-consuming traditional method in deep loops to a 4000× to 5000× improvement in computing speed, at the expense of a 2% to 3% intensity error due to grid snapping. The performance of the new algorithm was tested with several cases and the results with the improvement and error are consistent. Coding is very easy and straightforward on popularly commercially available platforms. Although there are also methodologies that can greatly save the computation time, such as the widely used SOCS, the method reported in this paper is more straightforward and can be easily adopted. SOCS needs to setup and solve the eigenvalues and eigenfunctions, whenever a new illumination or pupil function is adopted. This method only needs to set up the converting pixel and scaling factor once for all. It is very suitable to be adopted as the running engine of optimization problems. The potential of using this method for lithography process development is expected to be high.

Acknowledgments

This paper was funded by the Taiwan National Science and Technology Council with Project Number NSTC 111-2622-E-007-018.

Source Code and Data Access

The source codes and simulation data of this paper can be accessed through *Code Ocean* with the capsule "wavelength-scaling FFT."

References

1. J. P. Stirniman and M. L. Reiger, "Fast proximity correction with zone sampling," *Proc. SPIE* **2197**, 294–301 (1994).
2. N. B. Cobb, "Fast optical and process proximity correction algorithm for integrated circuit manufacturing," PhD dissertation, U. C. Berkeley (1998).
3. M. L. Rieger et al., "Enriching design intent for optimal OPC and RET," *Proc. SPIE* **4754**, 132–137 (2002).
4. N. B. Cobb and Y. Granik, "Model-based OPC using the MEEF matrix," *Proc. SPIE* **4889**, 1281–1292 (2002).
5. Y. Zhang et al., "A focus exposure matrix model for full chip lithography manufacturability check and optical proximity correction," *Proc. SPIE* **6283**, 62830W (2006).
6. J. R. Gao et al., "MOSAIC: mask optimizing solution with process window aware inverse correction," in *51st ACM/EDAC/IEEE Design Autom. Conf. (DAC)*, pp. 1–6 (2014).
7. H. J. Levinson, *Computational Lithography for EUV*, SPIE Press, Bellingham, Washington (2020).
8. A. Erdmann, *Optical and EUV Lithography: A Modeling Perspective*, SPIE Press, Bellingham, Washington (2021).
9. A. E. Rosenbluth et al., "Optimum mask and source patterns to print a given shape," *Proc. SPIE* **4346**, 486–502 (2001).
10. A. Erdmann et al., "Toward automatic mask and source optimization for optical lithography," *Proc. SPIE* **5377**, 646–657 (2004).
11. R. Socha, X. Shi, and D. Lehoty, "Simultaneous source mask optimization (SMO)," *Proc. SPIE* **5853**, 180–193 (2005).
12. R. Socha et al., "Design compliant source mask optimization (SMO)," *Proc. SPIE* **7748**, 77480T (2010).
13. J. Pomplun et al, "Reduced basis method for source mask optimization," *Proc. SPIE* **7823**, 78230E (2010).
14. N. Jia and E. Y. Lam, "Pixelated source mask optimization for process robustness in optical lithography," *Opt. Express* **19**(20), 19384–19398 (2011).
15. A. Poonawala and P. Milanfa, "OPC and PSM design using inverse lithography: a nonlinear optimization approach," *Proc. SPIE* **6154**, 61543H (2006).
16. L. Pang et al., "Inverse lithography technology (ILT): a natural solution for model-based SRAF at 45-nm and 32-nm," *Proc. SPIE* **6607**, 660739 (2007).
17. Y. Shen et al., "Level-set-based inverse lithography for photomask synthesis," *Opt. Express* **17**(26), 23690–23701 (2009).
18. C. A. Mack, "PROLITH: a comprehensive optical lithography model," *Proc. SPIE* **0538**, 207–220 (1985).

19. C. A. Mack, "Advanced topics in lithography modeling," *Proc. SPIE* **0631**, 276–285 (1986).
20. S. Sherwin et al., "Modeling high-efficiency extreme ultraviolet etched multilayer phase-shift masks," *J. Micro/Nanolithogr. MEMS, MOEMS* **16**(4), 041012 (2017).
21. J. Finders et al., "Low-k1 imaging: how low can we go?" *Proc. SPIE* **4226**, 1–15 (2000).
22. L. W. Liebmann, "Resolution enhancement techniques in optical lithography: it's not just a mask problem," *Proc. SPIE* **4409**, 23–32 (2010).
23. S. Y. Fang et al., "Simultaneous OPC- and CMP-aware routing based on accurate closed-form modeling," in *Proc. 2013 ACM Int. Symp. Phys. Des.*, pp. 77–84 (2013).
24. B. J. Lin, *Optical Lithography: Here Is Why*, 2nd ed., SPIE Press, Bellingham, Washington (2021).
25. C. Mack, *Fundamental Principles of Optical Lithography: The Science of Microfabrication*, John Wiley and Sons (2007).
26. J. W. Goodman, *Introduction to Fourier Optics*, Roberts and Company Publishers (2005).
27. N. B. Cobb et al., "Mathematical and CAD framework for proximity correction," *Proc. SPIE* **2726**, 208–222 (1996).
28. J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Math. Comput.* **19**, 297–301 1965).
29. E. O. Brigham, *The Fast Fourier Transform and Its Application*, Prentice Hall (1988).

Tsai-Sheng Gau received his PhD in physics at Taiwan Tsinghua University in 1994. He spent 2 years in the same university for the postdoctoral research on x-ray diffraction of synchrotron radiation, and then joined ITRI for 0.25 μm deep-submicron semiconductor research. He joined tsmc R&D lithography group in 2000. In tsmc, he implemented 193-nm immersion technology to production and he developed and delivered OPC technologies from 28 to 3 nm generations. He led R&D OPC and mask groups and delivered EUV mask technology before his retirement from tsmc in 2021. He is the distinguished professor in the college of semiconductor research in NTHU now.

Biographies of the other authors are not available.