Computation lithography: virtual reality and virtual virtuality

Edmund Y. Lam[a] and Alfred K. K. Wong

[a]Imaging Systems Laboratory,
Department of Electrical and Electronic Engineering,
The University of Hong Kong,
Pokfulam Road, Hong Kong

elam@eee.hku.hk

Abstract: Computation lithography is enabled by a combination of physical understanding, mathematical abstraction, and implementation simplification. An application in the virtual world of computation lithography can be a virtual reality or a virtual virtuality depending on its engineering sensible-ness and technical feasibility. Examples under consideration include design-for-manufacturability and inverse lithography.

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

References and links

1. F. H. Dill, W. Hornberger, P. Hauge, and J. Shaw, "Characterization of positive photoresists," IEEE Trans. Electron Devices **ED-22**(7), 445–452 (1975).
2. K. L. Konnerth and F. H. Dill, "In-situ measurement of dielectric thickness during etching or developing process," IEEE Trans. Electron Devices **ED-22**(7), 452–456 (1975).
3. F. Dill, "The basis for lithographic modeling," in *Proc. SPIE*, B. W. Smith, ed., vol. 5754, pp. 377–382 (2005).
4. C. A. Mack, "Thirty years of lithography simulation," in *Proc. SPIE*, B. W. Smith, ed., vol. 5754, pp. 1–12 (2005).
5. A. Neureuther, "If it moves, simulate it!" in *Proc. SPIE*, H. J. Levinson and M. V. Dusa, eds., vol. 6924, p. 692402 (2008).
6. M. Yeung, "Modeling aerial images in two and three dimensions," in *Proc. Kodak Microelectronics*, pp. 115–126 (1985).
7. D. Nyyssonen and C. P. Kirk, "Optical microscope imaging of lines patterned in thick layers with variable edge geometry: theory," J. Opt. Soc. Am. A **5**(8), 1270–1280 (1988).
8. K. Lucas, C.-M. Yuan, and A. Strojwas, "A Rigorous and Practical Vector Model for Phase Shifting Masks in Optical Lithography," in *Proc. SPIE*, J. D. Cuthbert, ed., vol. 1674, pp. 252–263 (1992).
9. T. Matsuzawa, A. Moniwa, N. Hasegawa, and H. Sunami, "Two-Dimensional Simulation of Photolithography on Reflective Stepped Substrate," IEEE Trans. Comput.-Aided Des. Int. Cir. Sys. **6**(3), 446–451 (1987).
10. H. P. Urbach and D. A. Bernard, "Modeling latent image formation in photolithography using the Helmholtz equation," in *Proc. SPIE*, V. Pol, ed., vol. 1264, pp. 278–293 (1990).
11. K. S. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media," IEEE Trans. Antennas Propag. **14**, 302–307 (1966).
12. R. Guerrieri, K. H. Tadros, J. Gamelin, and A. Neureuther, "Massively parallel algorithms for scattering in optical lithography," IEEE Trans. Comput.-Aided Des. Int. Cir. Sys. **10**(9), 1091–1100 (1991).
13. J. A. Sethian, "Fast marching level set methods for three-dimensional photolithography development," in *Proc. SPIE*, G. Fuller, ed., vol. 2726, pp. 262–272 (1996).
14. S. Osher and J. A. Sethian, "Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations," J. Comput. Phys. **79**, 12–49 (1988).
15. J. F. Chen, T. Laidig, K. Wampler, and R. Caldwell, "Optical proximity correction for intermediate-pitch features using sub-resolution scattering bars," J. Vac. Sci. Technol. B **15**(6), 2426–2433 (1997).
16. O. Otto and R. Henderson, "Advances in process matching for rules-based optical proximity correction," in *Proc. SPIE*, vol. 2884, pp. 425–434 (1996).

17. N. Cobb and A. Zakhor, "Experimental Results on Optical Proximity Correction with Variable Threshold Resist Model," in *Proc. SPIE*, G. Fuller, ed., vol. 3051, pp. 458–468 (1997).
18. M. Rieger and J. Stirniman, "Mask fabrication rules for proximity corrected patterns," in *Proc. SPIE*, vol. 2884, pp. 323–332 (1996).
19. T. Waas, H. Eisenmann, and H. Hartmann, "Proximity Correction for high CD-Accuracy and Process Tolerance," in *Proc. Symposium on Nanocircuit Engineering* (1994).
20. H.-Y. Liu, L. Karklin, Y.-T. Wang, and Y. C. Pati, "Application of alternating phase-shifting masks to 140 nm Gate Patterning II: Mask design and manufacturing tolerances," in *Proc. SPIE*, vol. 3334, pp. 2–14 (1998).
21. H. Gamo, "Matrix Treatment of Partial Coherence," in *Progress in Optics*, E. Wolf, ed., vol. 3, pp. 187–332 (North-Holland, 1964).
22. Y. C. Pati, A. A. Ghazanfarian, and R. F. Pease, "Exploiting Structure in Fast Aerial Image Computation for Integrated Circuit Patterns," IEEE Trans. Semi. Manufactur. **10**(1), 62–74 (1997).
23. A. E. Rosenbluth, G. Gallatin, R. Gordon, W. Hinsberg, J. Hoffnagle, F. Houle, K. Lai, A. Lvov, M. Sanchez, and N. Seong, "Fast calculation of images for high numerical aperture lithography," in *Proc. SPIE*, B. Smith, ed., vol. 5377, pp. 615–628 (2004).
24. M. Born and E. Wolf, *Principles of Optics*, pp. 491–555, sixth ed. (Pergamon Press, 1980).
25. J. W. Goodman, *Introduction to Fourier Optics*, 3rd ed. (Roberts and Company, 2004).
26. A. K. Wong, *Resolution Enhancement Techniques in Optical Lithography* (SPIE Press, 2001).
27. A. K. Wong, *Optical Imaging in Projection Microlithography* (SPIE Press, 2005).
28. J. Kim and M. Fan, "Hotspot detection on post-OPC layout using full-chip simulation-based verification tool: a case study with aerial image simulation," in *Proc. SPIE*, K. R. Kimmel and W. Staud, eds., vol. 5256, pp. 919–925 (2003).
29. S. D. Shang, Y. Granik, N. B. Cobb, W. Maurer, Y. Cui, L. W. Liebmann, J. M. Oberschmidt, R. N. Singh, and B. R. Vampatella, "Failure prediction across process window for robust OPC," in *Proc. SPIE*, A. Yen, ed., vol. 5040, pp. 431–440 (2003).
30. H. Mashita, T. Kotani, F. Nakajima, H. Mukai, K. Sato, S. Tanaka, K. Hashimoto, and S. Inoue, "Tool-induced hotspot fixing flow for high volume products," in *Proc. SPIE*, vol. 7028, p. 70283I (2008).
31. A. K. K. Wong and E. Y. Lam, "The Nebulous Hotspot and Algorithm Variability," in *Proc. SPIE*, vol. 7275, p. 727509 (2009).
32. J. H. Huang, Z. H. Lui, M. C. Jeng, P. K. Ko, and C. Hu, "A Robust Physical and Predictive Model for Deep-Submicrometer MOS Circuit Simulation," Master's thesis, University of California, Berkeley (1993). Memorandum No. UCB/ERL M93/57.
33. S. Banerjee, P. Elakkumanan, L. W. Liebmann, J. A. Culp, and M. Orshansky, "Electrically driven optical proximity correction," in *Proc. SPIE*, V. K. Singh and M. L. Rieger, eds., vol. 6925, p. 69251W (2008).
34. L. Pang, Y. Liu, and D. Abrams, "Inverse lithography technology (ILT): What is the impact to the photomask industry?" in *Proc. SPIE*, M. Hoga, ed., vol. 6283, p. 62830X (2006).
35. S. H. Chan, A. K. Wong, and E. Y. Lam, "Initialization for robust inverse synthesis of phase-shifting masks in optical projection lithography," Opt. Express **16**, 14,746–14,760 (2008).
36. A. Poonawala and P. Milanfar, "Mask design for optical microlithography—an inverse imaging problem," IEEE Trans. Image Process. **16**, 774–788 (2007).
37. N. Jia, A. K. Wong, and E. Y. Lam, "Robust Photomask Design with Defocus Variation Using Inverse Synthesis," in *Proc. SPIE*, vol. 7140, p. 71401W (2008).
38. E. Y. Lam and J. W. Goodman, "Iterative Statistical Approach to Blind Image Deconvolution," J. Opt. Soc. Am. A **17**(7), 1177–1184 (2000).
39. B. Yenikaya and A. Sezginer, "A rigorous method to determine printability of a target layout," in *Proc. SPIE*, A. K. K. Wong and V. K. Singh, eds., vol. 6521, p. 652112 (2007).
40. A. E. Rosenbluth, S. Bukofsky, C. Fonseca, M. Hibbs, K. Lai, A. F. Molless, R. N. Singh, and A. K. K. Wong, "Optimum mask and source patterns to print a given shape," J. Microlithogr., Microfabr., Microsyst. **1**(1), 13–30 (2002).
41. W. H. Arnold, "Guest Editorial: Special Section on Double-Patterning Lithography," J. Micro/Nanolith. MEMS MOEMS **8**(1), 011,001 (2009).

1. Introduction

The current prevalence of computation lithography traces its root to the 1970s, when it was discovered that photoresist dissolution can be modeled as a surface etching phenomena, with the etch rate dependent on the state of photochemistry rather than the manner in which the resist was exposed [1, 2]. However, it has taken some time before lithography simulation, the precursor of computation lithography, establishes its present-day credibility. (For readers interested in the history of lithography simulation, they are referred to the articles by Dill [3],

Mack [4], and Neureuther [5], which consider it from a few points of view.) Even as late as the 1980s lithography simulation was still labeled by some as "Yesterday's technology simulated tomorrow" [5].

Nevertheless, the 1970s and 1980s was a germinating period for lithography simulation; many fundamental algorithms were developed. Computation of partially coherent imaging was placed on a firm foundation [6], permitting accurate simulation of projection printing in the regime where neither the relationship between mask and intensity nor that between objects and image-plane field is linear. A variety of techniques, including the waveguide method [7, 8], and finite-element [9, 10] and finite-difference approaches [11, 12], were developed to examine scattering and diffraction of light propagation through mask openings. Resist development modeling, which had been plagued by looping instability in which surfaces of dissolved materials curve back onto itself [13], was properly handled by the fast-marching level-set method [14, 13].

These algorithms enhanced a lithographer's arsenal of tools, complementing empirical data supplied by wafer exposures. With exponential improvement in computer speed and rising equipment cost, simulation imperceptibly assumed an ever-increasing role in lithography process development. During the 1970s and 1980s, however, algorithm performance still restricted analysis to dimensions on the order of a feature size.

Adoption of optical proximity correction (OPC) in the 1990s triggered further innovation. Success of rule-based OPC [15, 16] quickly translated to a demand for model-based OPC [17, 18, 19, 20], which clamored full-chip image computation capabilities. Coherent decomposition [21, 22, 23] was equal to the challenge. By approximating a partially coherent optical system as a weighted collection of coherent systems, computation complexity was reduced by orders of magnitude.

2. Ingredients of computation lithography

Model-based OPC is an epitome of computation lithography, which can be defined as the use of computer algorithms in lithography-related integrated circuit (IC) development. The basic components of a computation lithography application are

1. specifications;

2. physical understanding;

3. mathematical abstraction; and

4. implementation simplification.

Let us illustrate these aspects using full-chip image computation as an example.

2.1. Specifications

The goal of full-chip image computation is to calculate the image intensities at select locations (or critical dimensions of select features) on a design quickly and accurately. Depending on the application, "quickly" and "accurately" are specified by concrete numbers such as 1 s per intensity value and 1 nm maximum deviation from empirical calibration data.

2.2. Physical understanding

In order to compute an image, a comprehension of image formulation principles is necessary. From an understanding of light diffraction, the physics of a projection microlithography system can be distilled, as illustrated in Fig. 1 [24, 25]. A quasi-monochromatic light source of mean wavelength λ illuminates the reticle via the condenser lens L_1. In conjunction with L_1, the lens
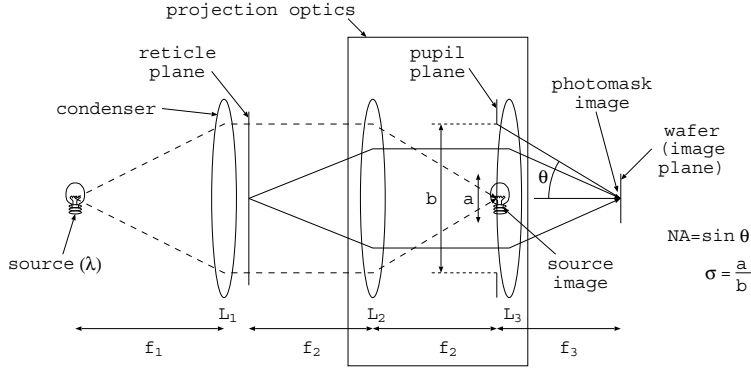
Fig. 1. Illustration of a projection microlithography exposure system [26].

L_2 forms an image of the source in the pupil. This lens L_2 performs a Fourier transform of the object such that the field across the pupil plane is the spectrum of the object. Low-spatial-frequency components pass closer to the center of the pupil, and higher-frequency components are nearer to the periphery of the pupil. The highest frequencies are cut off by the pupil. The lens L_3 forms the image by combining the filtered frequency components.

2.3. Mathematical abstraction

Before an algorithm for image computation can be formulated, operation of the exposure system depicted in Fig. 1 needs to be described mathematically. Image intensity at a point (x, y) is given by [27]:

$$I(x,y) = \iiiint\limits_{-\infty}^{+\infty} w(x-x'_o, y-y'_o; x-x''_o, y-y''_o) O(x'_o, y'_o) O^*(x''_o, y''_o)\, dx'_o dy'_o dx''_o dy''_o, \quad (1)$$

where

$$w(x'_o, y'_o; x''_o, y''_o) = J(x'_o - x''_o, y'_o - y''_o) H(x'_o, y'_o) H^*(x''_o, y''_o),$$

$$J(x'_o - x''_o, y'_o - y''_o) \quad \text{is the mutual intensity,}$$

$$H(x,y) \quad \text{is the optical system transfer function,}$$

$$O(x,y) \quad \text{is the mask transmittance,}$$

and the asterisk $*$ represents complex conjugation. Equation (1) says that the image intensity at (x,y) can be interpreted as the weighted sum (w) of contributions from neighboring objects $O(x'_o, y'_o)$ and $O(x''_o, y''_o)$.

2.4. Implementation simplification

Computation complexity of Eq. (1) is high because of the four-fold integral. To improve efficiency, the function w can be decomposed into a series of its eigenvectors φ_k and corresponding eigenvalues λ_k [21], such that

$$w(x'_o, y'_o; x''_o, y''_o) = \sum_{k=1}^{\infty} \lambda_k \varphi_k(x'_o, y'_o) \varphi_k^*(x''_o, y''_o).$$

For typical exposure systems, the magnitudes of a few eigenvalues are much larger than the rest. We can thus approximate the image as a finite sum of, say, K dominant eigenvectors:

$$I(x,y) \cong \sum_{k=1}^{K} \lambda_k \left| \iint\limits_{-\infty}^{+\infty} \varphi_k(x-x',y-y')O(x',y')\,dx'dy' \right|^2. \tag{2}$$

Although Eq. (2) reduces the four-fold integral of Eq. (1) to K two-dimensional convolutions, computation complexity is still high. Nevertheless, for layouts that can be decomposed exclusively into rectangles, we can represent each rectangle as the sum of four quadrant functions

$$Q(x,y) = \begin{cases} 1 & \text{if } x \geq 0 \text{ and } y \geq 0, \\ 0 & \text{otherwise,} \end{cases}$$

such that

$$O(x,y) = \sum_{n=1}^{N} t_{\text{fg}}^{(n)} \left[Q(x-x_1^{(n)}, y-y_1^{(n)}) - Q(x-x_2^{(n)}, y-y_1^{(n)}) + \right.$$
$$\left. Q(x-x_2^{(n)}, y-y_2^{(n)}) - Q(x-x_1^{(n)}, y-y_2^{(n)}) \right],$$

where N is the total number of rectangles, the superscript (n) indexes the rectangular shapes, and $t_{\text{fg}}^{(n)}$ represents the transmittance of the nth shape [22, 23]. Substituting the above expression into Eq. (2) results in

$$I(x,y) \cong \sum_{k=1}^{K} \left| \sum_{n=1}^{N} t_{\text{fg}} \left[\psi_k(x-x_1^{(n)}, y-y_1^{(n)}) - \psi_k(x-x_2^{(n)}, y-y_1^{(n)}) + \right. \right.$$
$$\left. \left. \psi_k(x-x_2^{(n)}, y-y_2^{(n)}) - \psi_k(x-x_1^{(n)}, y-y_2^{(n)}) \right] \right|^2, \tag{3}$$

where

$$\psi_k(x,y) = \sqrt{\lambda_k} \cdot Q(x,y) \otimes \varphi_k(x,y)$$

is the convolution of the quadrant function with the kth eigenvector scaled by the eigenvalue. Since the functions $\psi_k(x,y)$ are independent of the object, they can be precomputed and stored. Image calculation hence becomes an addition operation, improving computation efficiency tremendously.

3. Virtual reality and virtual virtuality

In the virtual world of computation lithography, virtual realities are those applications that perform useful service for IC development. They are tools characterized by

1. specifications that make engineering sense, and

2. performance (based on physical understanding, mathematical abstraction, and implementation simplification) that meets the specification targets.

Virtual virtualities, on the other hand, are algorithms that fall short of one or both of the two criteria. That model-based OPC and post-OPC verification [28, 29] are virtual realities there is little doubt; they are incumbents in a standard production flow. But it is of interest to examine the degree of reality of a few applications at the fringe.

3.1. Physical DFM

Physical design-for-manufacturability (DFM) refers to remedying weaknesses in a physical design [30]. It comprises finding hotspots by model-based verification, and fixing them by polygon movements. Since these two steps are rather straightforward, there is little implementation difficulty. The issue lies with specifications.

During the hotspot analysis stage in physical DFM, it is sometimes impractical, for turnaround or for accessibility reasons, to apply the same algorithm as the one used for sign-off verification. Differences between the DFM and the sign-off algorithms would result in hotspot discrepancy, leading to imperfect accuracy such as misses (hotspots not being remedied) and extraneous fixing (repairing of non-hotspots). The achievable accuracy depends on algorithm precision.

For example, suppose we conduct two hotspot analyses, the first one termed *reference* and the second *trial*. We can define the hotspot matching rate as

$$\mathscr{P}_{\text{matching}} = \frac{\text{number of common hotspots}}{\text{number of hotspots in reference}}$$

and the extra hotspot rate as

$$\mathscr{P}_{\text{extra}} = \frac{\text{number of trial hotspots not in reference}}{\text{number of hotspots in reference}}.$$

The missing hotspot probability is then

$$\mathscr{P}_{\text{missing}} = 1 - \mathscr{P}_{\text{matching}}.$$

With a critical dimension simulation variability of 5% ($1\,\sigma$) of the hotspot threshold for both simulators, it is computed that $\mathscr{P}_{\text{matching}} = 73.4\%$ and $\mathscr{P}_{\text{missing}} = \mathscr{P}_{\text{extra}} = 26.6\%$ respectively [31]. In addition, if we want to increase the matching rate by raising the trial threshold, the tradeoff is a significantly higher extra rate [31]. Thus, more stringent accuracy specifications than such theoretical bounds are unrealistic.

3.2. Electrical DFM

From a full-chip image lithography contours can be derived. By employing a suitable wire delay or transistor model such as BSIM [32], electrical performance can be calculated and fed back for OPC or layout enhancement. Such electrical DFM approach was shown to be viable from the computational standpoint [33].

Specifications for these applications, however, must consider all factors that significantly affect electrical behavior. An example is the possible interaction between corner-rounding effects and oxidation-induced stress effects. Should stress effects be found to be of higher relative importance, contour-to-electrical DFM may not make full engineering sense.

3.3. Inverse lithography

Inverse lithography aims to derive the optimal mask given a desired image [34, 35]. Since the synthesized mask is unconstrained by the topology of the original physical design, inverse lithography promises to deliver superior performance over existing model-based OPC approaches. The specifications of inverse lithography make engineering sense, for they are essentially those of OPC's.

However, current performance still falls short, primarily because of the lack of a mathematical description for mask regularization [36] that provides suitable control of lithography robustness and mask manufacturability. We illustrate this with two ways of computing a binary

(a) target pattern (b) mask

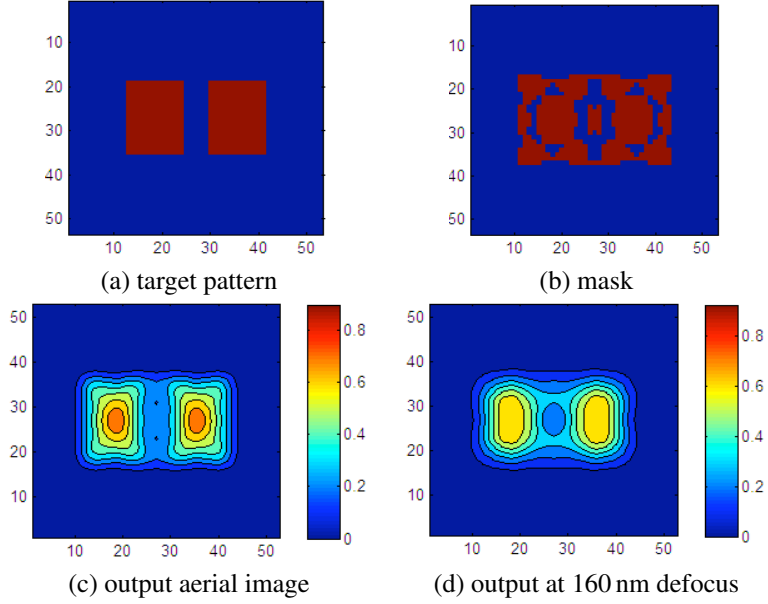(c) output aerial image (d) output at 160 nm defocus

Fig. 2. An example of inverse lithography with no regularization.

mask. Mathematically, the pixels of the mask can take on two possible values, namely $\{0,1\}$. Assume we use a "sigmoid" function to approximate resist effect (where the variable a adjusts the contrast, and t_r represents the threshold) [36], we find an optimal mask by minimizing the mean square error (MSE), i.e.

$$O_{\text{opt}}(x,y) = \underset{O(x,y)\in\{0,1\}}{\arg\ \min} \sum_{x,y} \left[\frac{1}{1+\exp\left\{-a\left(|H(x,y)\otimes O(x,y)|^2 - t_r\right)\right\}} - \hat{I}(x,y) \right]^2 \quad (4)$$

where $O(x,y)$ is the mask transmittance, $O_{\text{opt}}(x,y)$ is the optimal mask, $\hat{I}(x,y)$ is the desired circuit pattern, and $H(x,y)$ is the optical transfer function (OTF) of the imaging system. Eq. (4) can be considered an *unconstrained* optimization. Fig. 2 shows an example of this computation. The target pattern is given in (a), and the unconstrained mask design is given in (b). The output aerial image is given in (c). By thresholding this image, we can obtain a binary circuit pattern that is very close to the target pattern. However, this design has a problem if the imaging parameter is not accurate. Consider, in particular, that the imaging system has 160 nm defocus. As shown in (d), the aerial image has the two rectangles merged together, and no thresholding can bring the pattern close to the original design.

On the other hand, we can specifically design our mask to be robust against focus variation. The full detail of the derivation is presented at [37], which is summarized below. Assume β is a Gaussian distributed focus error. We can append the quadratic phase factor $\exp\left\{-j\pi(f^2+g^2)\beta\right\}$ to the Fourier transform of $H(x,y)$, and optimize instead

$$O_{\text{opt}}(x,y) = \underset{O(x,y)\in\{0,1\}}{\arg\ \min}\ \mathscr{E}_\beta\left\{ \sum_{x,y} \left[\frac{1}{1+\exp\left\{-a\left(|H(x,y)\otimes O(x,y)|^2 - t_r\right)\right\}} - \hat{I}(x,y) \right]^2 \right\},$$

$$(5)$$

(a) target pattern   (b) mask

(c) output aerial image   (d) output at 160 nm defocus
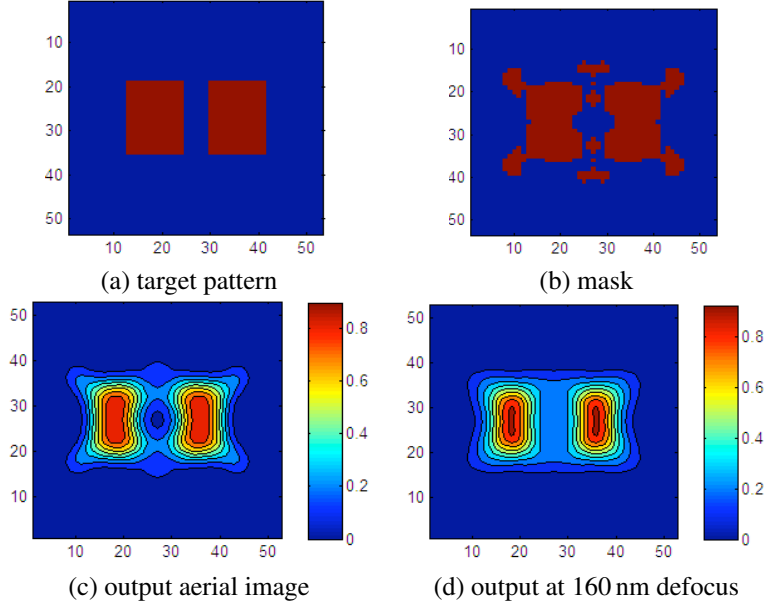
Fig. 3. An example of inverse lithography designed for robustness against defocus.

where $\mathscr{E}_\beta$ denotes expectation over the variable β. We call this a mask design regularized for robustness against focus variation. Note that the regularization is achieved implicitly rather than with an explicit penalty term, as in a typical image reconstruction and restoration algorithm [38]. Fig. 3 shows the mask design for the same pattern as before, but using this new methodology. We can see in (b) that the resulting mask design is very different from that in Fig. 2. It is, in fact, closer to what we normally expect with OPC. In (c), the output aerial image at nominal focus value is good in the sense that proper thresholding would result in a pattern as designed. Furthermore, it has sharper edges, making it a more robust design. In particular, as shown in (d), the design is still useful even at 160 nm defocus, as we can still separate the two rectangles by thresholding at around 0.5.

The above illustrates the dilemma in inverse lithography: On one hand, the flexibility of mask design leads to the potential of better solutions, yet on the other hand, proper control is necessary to make the mask more robust. Another important point to note is that both Fig. 2(b) and Fig. 3(b) do not represent very good designs in terms of mask manufacturability. To improve on this, one either needs to impose further constraints during the optimization process, or attach a postprocessing to the mask designs to make them more manufacturable. Both are still open problems to be solved for inverse lithography. In a similar manner, determining printability of a layout using a generic model [39] is hindered by deficiency of the mask model, and concurrent source-mask optimization [40] needs to resolve both mask and source regularization issues.

4. Remarks

If we consider a plane with the *x*-axis denoting the degree of engineering sensible-ness and the *y*-axis representing performance requirement (defined by some combination of turn-around, accuracy, and precision metrics), we can demarcate the plane into two areas. The one to the lower-right of the dividing line is the realm of virtual reality, while the region to the upper-left is the region of virtual virtuality. Applications can be placed onto this plane according to their
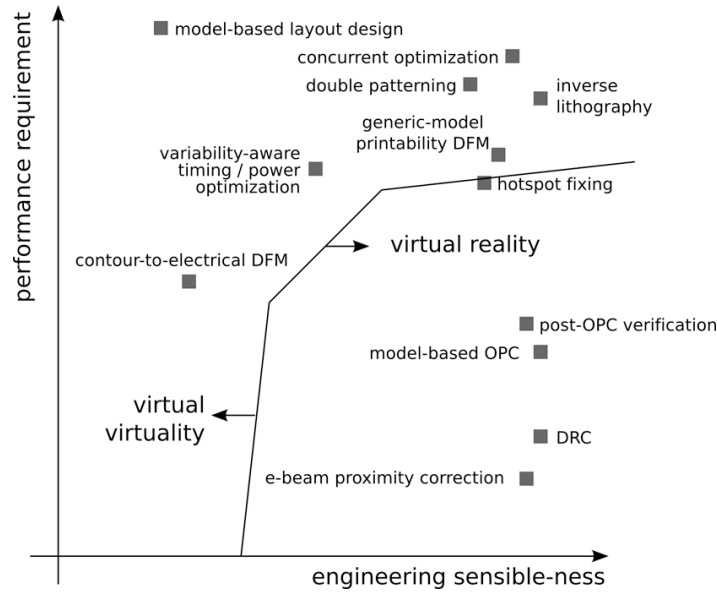
Fig. 4. The reality-virtuality space of computation lithography (not to scale).

degrees of reality, as illustrated in Fig. 4.

With algorithmic innovation and increased parallelization, the boundary between reality and virtuality will evolve, and the markers also travel on their own trajectories. Today's virtuality can become tomorrow's reality. For example, we put "double patterning" in the realm of virtuality, primarily because of its performance requirement. While there are currently device makers using double patterning technology to manufacture electronics devices, particularly memory devices, our view is that it is still not quite ready for full-scale manufacturing with all the necessary infrastructure: Electronic design automation software that performs seamless decomposition of physical designs, and manufacturing tools and processes that deliver the needed throughput and critical dimension or overlay control. However, it is getting close (see, for example, the January–March 2009 issue in *Journal of Micro/Nanolithography, MEMS, and MOEMS* with a special section on double-patterning lithography [41]).

Indeed, for something to be a reality, both engineering specifications need to make sense and the performance needs to meet the specifications. For applications such as double patterning and inverse lithography, the performance still needs improvement. For other applications such as physical DFM, the specifications may need to be revised, such as giving up on the 100% hotspot fixing requirement.

Simultaneously, with enhancements in our understanding and as requirements scale with technology, the markers may also travel in the reverse direction. For example, lithography model accuracy needs to improve with decreasing critical dimension, pushing the "model-based OPC" marker upwards in the graph. Today's reality may become tomorrow's virtuality.

For computation lithographers, the goals are to enlarge the virtual reality region and to push the markers towards the lower-right. In this regard, applications on the right side of Fig. 4 around the reality envelope deserve special attention.

Acknowledgment