



Real-time profile shape reconstruction using dynamic scatterometry

Sébastien SOULAN^a, Maxime BESACIER^a, Patrick SCHIAVONE^a, Laurent VALLIER^a,
Tanguy LEVEDER^a

^aLTM - CNRS, 17 av. des martyrs, 38054 Grenoble Cedex, France

^bCEA - LETI, 17 av. des martyrs, 38054 Grenoble Cedex, France

ABSTRACT

In-line process control in microelectronic manufacturing requires real-time and non-invasive monitoring techniques. Among the different metrology techniques, the scatterometry, based on the ellipsometric signatures (i.e. stokes coefficients vs. wavelengths) of light scattered by a patterned structures, seems to be well adapted to address this kind of problem. For instance, the shape evolution of a grating in real time during etch processing can be monitored. Traditionally, the direct problem of defining the shape and computing the signature is dealt with modal methods. However, the inverse problem can not be solved as easily. Different classes of algorithms have been introduced (evolutionary, simplex), but the method of library searching is more commonly used in industry. The main limit of this method is the acquisition time of data for different wavelengths. Then, a lack of data can leads to the method failure and several database patterns can be matched to the experimental data. In this article, a technique for real time reconstruction of grating shape variation using dynamic scatterometry is presented. The different tools to realize this reconstruction, such as Modal Method by Fourier Expansion, regularization technique and mostly the specific soft and hardware architecture are also developed. Results of a dynamic experiments will illustrated these presentations.

Keywords: Scatterometry, TODO

1. INTRODUCTION

A part of future advances in microelectronic is conditioned by the capability for industry to monitor the manufacturing process of their chips. Since the dimensions of transistors become smaller and smaller, an important effort has to be made in the field of metrology. CD metrology techniques must be able to characterize very short dimensions with a good accuracy. AFM and XSEM techniques are suitable to reach these accuracies, but these are invasive or long time consuming and actually, not well adapted to the industry requirements. Indeed, the next step in the metrology improvement is the development of a technique able to follow the evolution of a pattern during a dynamic process. It means that the metrology tool must be placed in situ, the time of acquisition of data must be very fast and this technique must be non invasive. This objective is not a simple task but first experimental tests can be carried out for simplest processing (as described below). Based on the ellipsometric measurement of light diffracted by a grating, the scatterometry technique appears to be promising for in line process monitoring. The scatterometry is an optical and a non-invasive technique of metrology. It measures the change of polarization subsequent to the scattering of a ray of light. The used ellipsometer gives signatures for a single angle but for several wavelengths. The signatures resulting of the measurement are compared with simulated signatures coming from modelling for the best match. The simulated signatures are achieved by the Modal Method by Fourier Expansion starting from a given topography. This is the direct problem. On the opposite, the inverse problem consists in finding the shape of the grating thanks to the signatures and this is not a simple task. Different classes of algorithms have been developed (evolutionary, simplex),¹ but the method of library searching is practically the more commonly used in industry. Moreover, this method is real-time compliant as the solving time is perfectly predictable. The work presented in this article will deal with improvements made on this technique for the case of dynamic scatterometry. A signature library is built-up using different combinations of profile parameters (CD, height, sidewall angle, etc) and the experimental signature is compared with the library for the best match. Although generating the library requires a lot of computation time, solving the inverse problem can be computed quickly because pattern matching is reduced to a simple numerical database search. The constraint is the limit in wavelength resolution. Indeed, in order to follow the real time process evolution, the acquisition of data is limited. This poor resolution in wavelength can leads to matching problems (i.e. several database can be matched to the experimental data). In this paper, we present a technique which does not need to solve the full inverse problem at each time step but which keep a good accuracy on results. The main idea is to achieve a first and complete signature measurement for the whole scale of wavelengths at $t=0$ before processing, and partial signatures at next steps to intelligently refine the library search using Tikhonov time regularization. This technique assumes that only a smooth variation of the shape over time is realistic. This approach is implemented as a general C++ architecture dedicated to scatterometry. The system integrates multiprocessing (shared memory) capability for high-speed signature generation and the use of the Graphic Processin Unit (GPU, video card processor) as a numerical database accelerator for the library-based inverse problem. Due to the inherent vector architecture of the GPU, the entire library search operation can be made in only a few clock cycles. In a short first part, a brief description of the direct problem is presented. The second part deals with the real time regularisation description and the third part describes the software and the hardware tools. Finally, the first results are shown with a specific example. This application deals with the time evolution of the shape of a pattern resist when heated.

2. DIRECT PROBLEM SOLVING

As mentioned in the introduction part, the direct problem is used to establish simulated signatures from a given topography. It allows building the data library with parameter variations. Some modelling tools are suitable to address the direct problem. In this work, the Modal Method by Fourier Expansion is chosen. A full description of this method can be found elsewhere and will not be detailed here. Only a brief description is given. The Modal Method by Fourier Expansion is well adapted for the topography of the scatterometry problem.²

MMFE is based on a simple approach where Maxwell equations are solved thanks to a matrix system. TE or TM polarization are treated using the same formalism and an improved version allows to obtain good results for TM polarization in terms of convergence speed. In our study only the TE polarization has been addressed. This method implicitly considers periodic structures. The solution of Maxwell equations is reduced to the solution of an algebraic eigenvalue problem in a discrete Fourier space. The derivation of the matrix operator involves two steps: the electromagnetic field is expanded into Floquet-Fourier series, the permittivity of the materials which appears in the Maxwell equation is here a periodic function and is expanded into Fourier series. The topography of the structure which is studied is cut in different layers. These different layers are linked by a S matrix system which accounts for the boundary condition providing a good convergence even for thick or absorbing layers. In this rigorous method, the only approximation is due to the number of modes retained for the truncation of the modal expansion. In EUV lithography, due to the short wavelength, the ratio λ/d is very small. Consequently the number of mode is important according to the following equation:

$$\sin(\theta_m) = \sin(\theta) - m \frac{\lambda}{d}$$

where θ_0 is the incident angle, m is the number of mode and θ_m is the angle of the mode m .

This large number of mode leads to a compromise between computation time and results accuracy. In practice, to address complex topographies and for a retained number of mode of 51, the precision is good and the computation of the reflected field is performed in around 20 seconds with an Intel Xeon 3.4GHz processor. For the scatterometry problems, the number of mode must not be important in order to generate, as fast as possible, the database.

Typical library computation time, depending on m . The database contains 65.636 signatures, computed for 28 wavelengths with a grating shape divided in 32 layers. The library size in memory is 40MBytes.

$m = 0$	$m = 1$	$m = 2$	$m = 3$	$m = 10$
1 h	3h	5h30	10h	5d

Depending on the number of layer : (for $m = 3$)

$n = 32$	$n = 22$	$n = 12$
10 h	6h30	3h30

For a fixed m and n , the computation time and library size are linear, with respect to the number of wavelength for a signature and the number of signature contained in the library.

It's clear that computation time has to be dramatically reduced. We are working to make the code run on multiprocessor systems. Since a library is nothing but a collection of signatures that can be computed independently, the parallel efficiency is close to 100%. We have verified that on a quad-Intel Xeon computer, using an OpenMP library.

3. REAL-TIME REGULARIZATION

Dynamic scatterometry can not be reduced to successive static problems : a high frequency acquisition and a short acquisition time (relatively to process characteristic time) are required to follow the shape variation. Thus, standard ellipsometers that usually gives full defined signature in seconds are not adapted. However, we can use multi-way ellipsometers : ellipsometric signals can be acquired quickly but only for a selected set of wavelength. The number of wavelengths is low (so that a mere static scatterometry resolution is inefficient), but the grating shape can be reconstructed if we take into account the whole set of partial signatures and the regularity (the smoothness) of the shape parameters (height, CD, side wall angle, etc.). The principle of regularization can indeed be summarized as a balance for the choice of the best parameters. This balance is made between, on an hand, the proximity of each acquired signature to the chosen signature in the library and on an other hand, the regularity (smoothness) of the parameters variation over time.

Practically, the way we proceed is :

- at $t = 0$, i.e. before the process, the grating shape is supposed to be perfectly known. For that we can use, for example, static scatterometry, where a full defined signature is acquired. There is no problem of computing time saving here.
- then, at each time step, we acquire a low-resolution and low-precision signature. This implies bad accuracy in database look up. The technique of regularization begin with this : we don't pick up only one Nearest Neighbor (NN) to the acquired signatures in the database but the 4 nearest neighbors. So, even if the first NN is far to be the good solution (it can be an artifact), we assume that a better solution is included in the 4 NN. The algorithm of regularization whom results are presented here selects at each time step (but relatively to the previous time steps) the best set of parameters. A standard Tikhonov-based regularization³ is then computed :

$$\vec{X} \min \left(\left\| \vec{X} - \vec{X}_{exp} \right\|^2 + \beta \int_{window} \vec{X}'^2 dt \right) \text{ where } \vec{X} = (H, CD, sidewall \text{ angle} \dots)$$

The goal is to have, for each parameter describing the grating shape, a smooth and realistic variation.

Notice that, for the full regularization (integrating the "smart" library search), only two parameters have to be optimized.

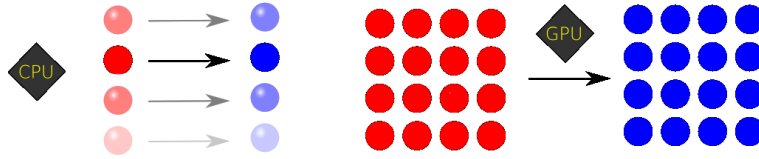


Figure 1. CPUs. vs. GPUs (S.I.M.D.)

4. HIGH PERFORMANCE COMPUTING

We've shown that the library search method is the only way to solve the inverse problem taking into account the "real-time" constraint. But the library used for this method is quite large : about half a GByte is a common size for industrial production. Large quantity of data implies two challenges :

- The computing time of this library should be improved at maximum; on a single (but powerfull : Xeon 3GHz) core, such a computation can last several weeks.
- Once computed, dealing with this data for quick library look-up is also a challenge : finding the Nearest Neighbor of an acquired signature inside a huge signature library should be made fast enough comparing to acquisition frequency of experimental data.

For the first point, we are working to make the code that computes libraries run on multiprocessor systems. Since a library is nothing but a collection of signatures that can be computed independently, the parallel efficiency is close to 100%. We have verified that on a dual dual-core Intel Xeon computer, using an OpenMP library.

For the second point, we use as a co-processor the GPU (Graphics processing Unit) of our graphic card. Computer game industry has a growing need to 3D graphics capability. Nowadays, video game picture rendering is made by a special stream processor, very powerfull but very different from standard CPUs. Indeed, GPUs are said to be stream or S.I.M.D. (**Single Instruction, Multiple Data**)processor because of their ability to compute a whole set of data (like a picture) with a single instruction. Contrary to CPUs, who, in their simplest form compute one data by instruction. For comprehensive explanation of what can be done efficiently with a GPU, read the paper.⁴

Using this vector processor is a bit tricky but can be very efficient for certain class of problem, the vector architecture however make it perfect for library look up. Finally, if we enumerate the advantages we have :

- For the problem of large library look-up, GPU are far faster than CPU.
- Main CPU is discharged from such computation and can be used for the second step of regularization.
- The power of GPU is increasing far fastest than GPU (see bellow)
- A graphic card with the most powerfull GPU of the moment only cost 700\$...

We have said that using GPU is a bit tricky because these processors are made only for graphics. If we want to compute data with, we need to think the signature database as a collection of big pictures in video memory. And computing in database (for instance, the calculation of the distance between the acquired signature and the pictures in the database) can be view as a visual effect to be applied to the plan where the picture is mapped. The visual effect is made by a lot of sub-processors inside the GPU core called *shaders*. In recent GPUs, shaders are programmable and we use this to develop a technique for multi-NN search. This technique is based on the work of Benjamin Bustos et al.⁵

Results Here we present some results for our problem, obtained with the NVidia GeForce 8800GTX GPU.⁶

The interest of using GPUs over CPU is clear on this chart. For large signature libraries (400 - 700 MB), GPU make the work about 10 times faster and even more than CPUs.

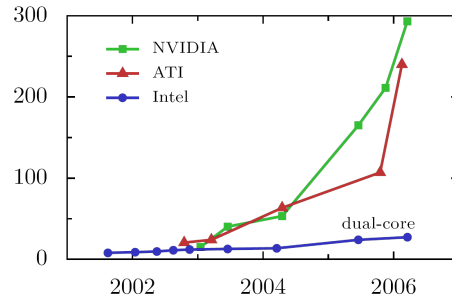


Figure 2. GPUs power, in here, GFLOPS increases far faster than CPUs power.

$N_{signature} \backslash N_{\lambda}$	5	10	15	25	50	100	150	200	250
16384						1.18	1.48	1.7	1.9
65536				1.5	2.61	4.31	5.64	6.41	7.18
262144	1.08	1.84	2.60	3.95	6.11	10.21	16.37		
1048576	1.70	2.92	4.04	6.18					

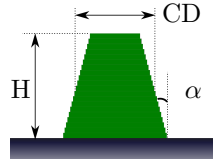
An important point is that using GPUs allow us to reconstruct grating shape *in time* ; fast enough regarding to device acquisition frequency. Using GPUs is a way to solve 10 or 20 signature frame per second, while regularizing...

5. RESULTS AND COMMENTS

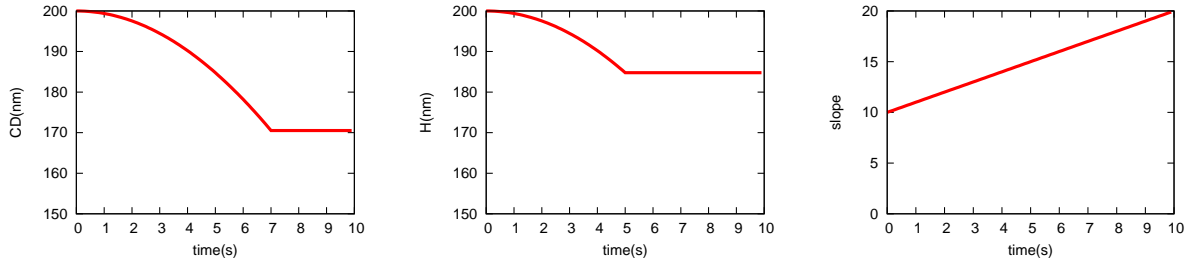
Here we show two examples of application for dynamic scatterometry where real-time regularization is used. The first one is for demonstration purpose: we solve the inverse problem on simulated data (i.e.. solution of the direct problem.). The second example uses a real set of scatterometry signature coming from a 32 way ellipsometer. The process we want to monitor is a reflow of patterned resist. Both of these experiment will allow us to make remarks and comments.

5.1. Results for simulated data

In this simulated experiment, we want to monitor a grating shape of which geometry is modeled by three parameters : the height H , the CD (*critical dimension* the width) and α the side wall angle.



Simulated movie Using our MMFE engine,² we simulate an "ellipsometric movie" of 100 frames (the process control lasts 10s and we use 10 frames per second). During this movie, the values of the three parameters are supposed to vary as such :



These parameter variation are arbitrary but includes successively a smooth part, a corner and then a flat part.

For the signature computation using the MMFE engine, each grating is modeled as a stack of 10 resist layers (neb22), lying on a silicon substrate. The angle of incidence of the light to be diffracted is set to 70. The truncate order for Fourier expansion is set to 10. These settings will be the same for the library computation.

Once the movie is generated, we add a noise to each frame (each signature) $S'_1 = (1 + \theta_1)S_1$, $S'_2 = (1 + \theta_2)S_2$ where $\theta_1, \theta_2 \in [0; 0.1]$ (5%) in order to take into account all the uncertainties due to modelization, measure and computation.

Library We choose to generate a signature library corresponding to the three-dimensional parameter space. Values taken by the parameters are :

- CD=160.32 to CD=200, by steps of 1.28
- H=160.32 to H=200, by steps of 1.28
- $\alpha=9.5$ to $\alpha=20$ by steps of 0.7

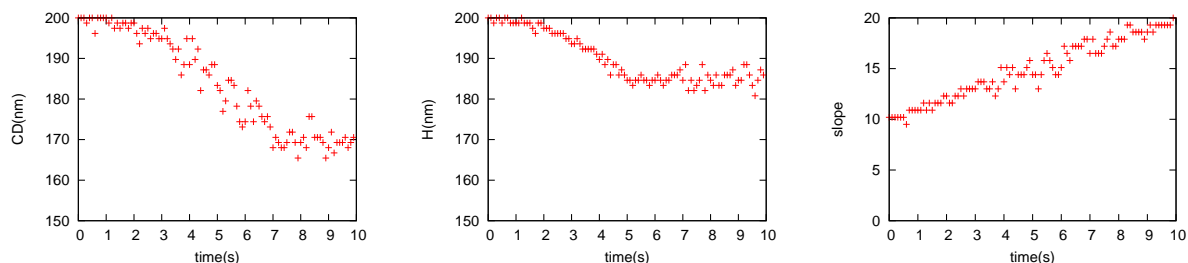
The choice of parameter range and density is the most tricky part in library computation : the library size in memory is proportional to the number of parameters combination. So, a good modeling work in necessary, first, to maintain the dimension of parameter space as low as possible, and then to restrict the

number of parameters values. Here we have $32 \times 32 \times 16 = 16384$ signatures in the library. Each signatures represents 16 wavelengths. Library size in memory, given by :

$$S_{Bytes} = N_{signature} \times 2N_{\lambda} \times \text{sizeof(float)}$$

is here of 4MBytes.

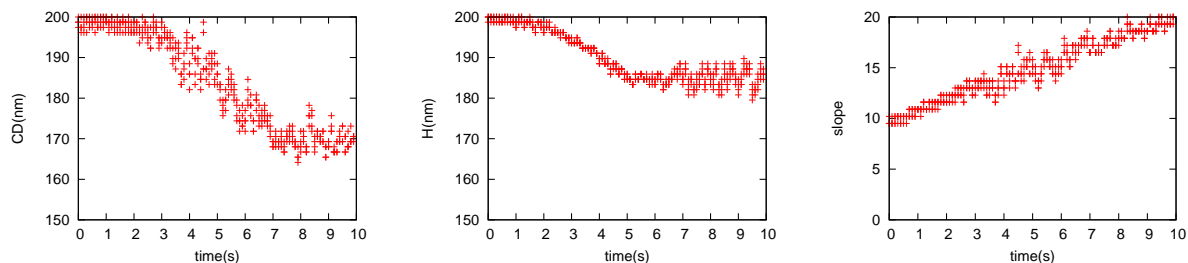
Dynamic reconstruction The last step of this experiment is to feed the program at each time step with simulated frames from the movie. You can see bellow the geometrical parameters set of the grating shape as it has been found using step-by-step static scatterometry using a mere library search.



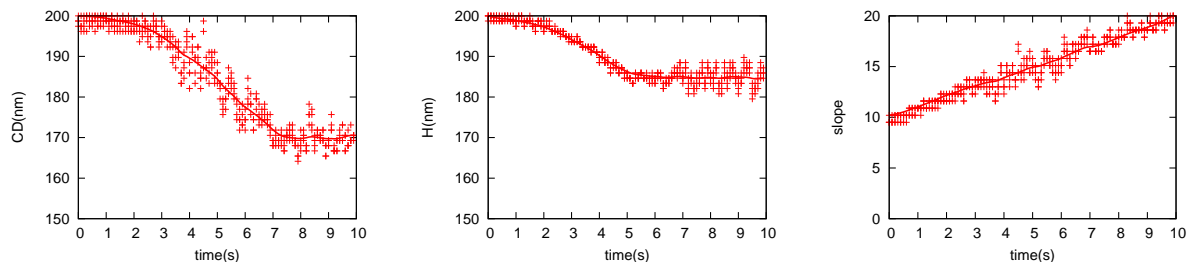
Geometrical parameters appears to be the most part of the time irregulars and sometimes false. The best match in library search is not necessarily the real solution : when parameters density is poor (for instance, because the modeling is difficult and thus, parameter space dimension is important); the solution vector \vec{X} can be close to real shape only by some of its component. For example, the raw library search method could privileges a solution with good H and CD and insignificant α than a more significant solution where the three parameters are closer to the reality.

Thus, the reconstruction process begins with a deeper library search where not only the first nearest neighbor is picked up, but also the three next. With four nearest neighbors at each time step, we can add another criterion for selection : the criterion of proximity to the data is kept, but it is now pondered by the regularity of the parameters. It's a way to throw away "bad" solutions coming from the library. This is the pre-regularizing step. **Un peu trop confidentiel non?**

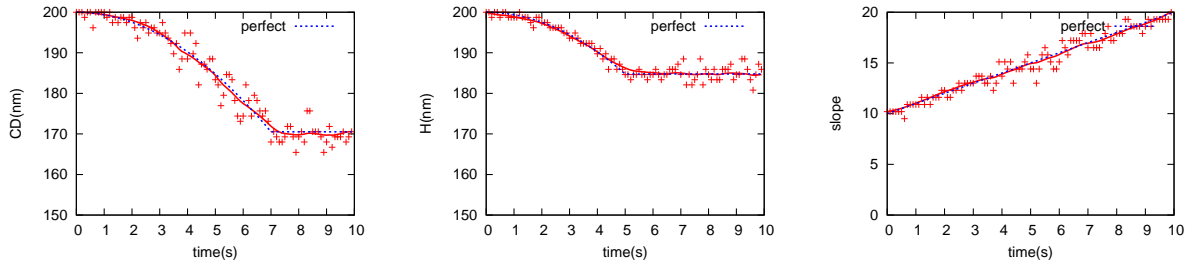
Picking-up the 4 nearest neighbors of each acquired signature (each frame) expands the field where realistic solution can be found:



Refining again with a standard Tikhonov³ time-based regularization gives :



Then a comparison with "original" simulated parameters gives :



Notice that the plus symbols are sometimes far from the solution, that is why a process of regularization is needed to reconstruct realistic shape variation.

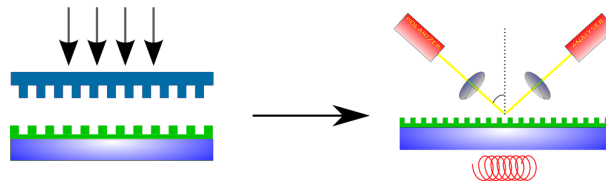
The principle of regularization poses a problem when the first derivative is supposed to be discontinuous-continuous : corners are softened. This can't be avoided but can be limited either by taking β lower as possible, or increasing frequency of acquisition.

We can see here the fundamental difference between these two regularizations : the first one does not make regular a parameter in itself (it can even be less regular than before), however, it regularizes the whole set of parameter

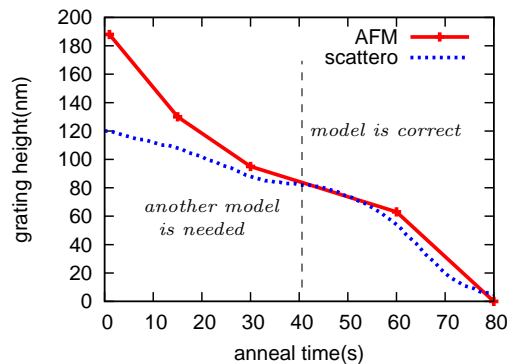
5.2. Resist pattern reflow

This experiment, originally carried out for resist viscosity study is here a way to begin experimental validation for our algorithms.

A nano-imprinted grating, made of resist on silicon is heated while scatterometry signatures are acquired. The hot-plate and the 32 wavelengths ellipsometer are from Horiba/Jobin Yvon. We made the temperature increase until reflow. (~ 90 deg C) and we try to watch pattern shape variation while reflowing.



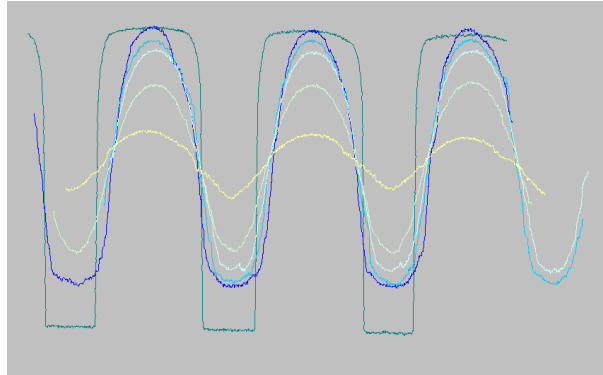
We first chose to model pattern shape using a mere sine wave : the amplitude is used as the only parameter. When we compare reconstructed profile shape with AFM measurements made in the same condition and at different time step, we get :



We can clearly see that there are two parts :

- A part, beginning at 40s where our simple model seems to work.
- And an other part, in the beginning of the reflow, where actually another model is required.

This is obviously because the grating shape is rectangular during the first part of the process, as we can see below :



This is typically a problem of scatterometry : when a good modeling has been found, standard scatterometry is efficient and dynamic shape reconstruction can work.

6. SUMMARY AND CONCLUSION

REFERENCES

1. C. J. Raymond, M. E. Littau, A. Chuprin, and S. Ward, "Comparison of solutions to the scatterometry inverse problem," in *Metrology, Inspection, and Process Control for Microlithography XVIII. Edited by Silver, Richard M. Proceedings of the SPIE, Volume 5375, pp. 564-575 (2004).*, R. M. Silver, ed., pp. 564–575, May 2004.
2. D. Fuard, C. Perret, V. Farys, and C. G. and Patrick Schiavone, "Measurement of residual thickness using scatterometry," **23**(6), pp. 3069–3074, AVS, 2005.
3. A. Tikhonov and V. Arsenin, *Solution of Ill-posed Problems*, 1977.
4. I. Buck, N. Govindaraju, M. Harris, J. Kruger, A. Lefohn, D. Luebke, T. Purcell, and C. Woolley, "Gpgpu: General-purpose computation on graphics hardware," in *Course 32 at ACM SIGGRAPH*, 2004.
5. S. H. Benjamin Bustos, Oliver Deussen and D. Keim, "A graphics hardware accelerated algorithm for nearest neighbor search," in *Computational Science – ICCS 2006*, P. M. S. Vassil N. Alexandrov, Geert Dick van Albada and J. Dongarra, eds., *LNCS 3994*, pp. 196–199, Springer, 2006.
6. <http://www.nvidia.com/page/geforce.8800.html>.