

# Data Evaluation as Source For Modelling in Nano-Imaging

B. Alles, B. Simeon, Technical University Munich  
E. Cotte, Advanced Mask Technology Center Dresden

Corresponding Author: B. Alles  
Advanced Mask Technology Center Dresden  
Raehnitzer Allee 9, 01139 Dresden, Germany  
Phone: +49/(0)351/4048-397  
E-Mail: benjamin.alles.external@photomask.com

## Abstract

In nano-imaging errors in positioning are crucial to the functional efficiency of the chips. Starting with this problem, a method for data evaluation based on smoothing splines is proposed. This method allows to distinguish between globally smooth systematic fingerprints and statistical noise. The results might be verified by a statistical test, like the D'Agostino-Pearson test. Then, a method for improving positioning performance is proposed which is solely based on the presented smoothing methods. Finally, first results of an appropriate simulation are shown.

## 1 Registration and Overlay

Modern chips, like computer CPUs or DRAMs, consist of twenty to forty structured layers. Those structures make up the electronic circuit. To make more powerful chips, manufacturers have to shrink features. As a result, the functional efficiency becomes more and more sensitive to placement errors between two consecutive layers. This error is called *overlay error*. Due to this issue, overlay specifications are tightened with each new generation of chips, see [1]. Since masks are blueprints of the layers' structures, positioning accuracy requirements on masks are becoming more rigid. Usually, the difference between the reference position and the actual position on the mask is examined. This concept is called *registration*. Today mask shops are searching for ways to improve the registration behaviour of their masks.

Mathematically speaking, registration (or overlay) is a function  $\text{reg}$  which assigns to every point  $(x, y)$  of the mask's front side  $\mathcal{M} \in \mathbb{R}^2$  its deviations  $(dx, dy)$ :

$$\begin{aligned} \text{reg} : \mathcal{M} &\rightarrow \mathbb{R}^2 \\ (x, y) &\rightarrow (dx, dy). \end{aligned}$$

Since only a finite number of points are measured, a registration (or overlay) data set consists of four vectors:  $\mathbf{x}, \mathbf{y}$  for the position of the measurement points and  $\mathbf{dx}, \mathbf{dy}$  for the corresponding deviation values. Details on overlay and registration can be found in [2].

## 2 Smoothing Splines

For correct interpretation of measured data, it is necessary to distinguish between statistical noise and systematic behaviour. Once a systematic fingerprint is identified, one might be able to correct for it. Statistical noise on the other hand cannot be corrected that easily: new tools, new measurement techniques or completely different approaches might be needed.

Statistical noise (e.g. stemming from measurement tools) varies with each measurement arbitrarily. To separate a deterministic part, one could repeat the measurement many times and average the data, which is the best way to find a systematic fingerprint. But sometimes multiple measurements cannot be performed, e.g. due to lack of measurement time or too high effort. In these cases another approach is needed:

In general, systematic fingerprints can be represented by smooth functions on the whole measurement area. Noise, on the other hand, is at least partially non-smooth, due to its arbitrary variation in each

measurement location. This difference is used by the smoothing spline approach to separate noise from deterministic behaviour.

Of course, this approach is not as good as averaging multiple measurements, since it cannot distinguish between non-smooth local systematic fingerprints and statistical noise. It also fails when only few data points are available, since the statistical noise might not induce enough roughness.

In the following sections, the construction of smoothing splines — first the one-dimensional case, then the multi-dimensional — based on [3] is summarized.

## 2.1 Interpolating Spline — one-dimensional

Suppose a given set of data points  $(x_i, z_i)_{i=1, \dots, n}$  with  $a = x_0 < x_1 < x_2 < \dots < x_n = b$ . Then one can define the corresponding cubic spline  $\mathcal{S}$  by the following properties:

- Restricted to each interval  $(x_{i-1}, x_i)$  for  $i = 1, \dots, n$ , the spline  $\mathcal{S}$  is a cubic polynomial.
- On the interval  $(a, b)$ , the spline  $\mathcal{S}$  is twice continuously differentiable. Mathematically speaking:  $\mathcal{S} \in \mathcal{C}^2((a, b), \mathbb{R})$ .

There are still two degrees of freedom undetermined, which are defined for different types. In this paper only *natural* cubic splines are considered. The additional claim is in this case:

$$\left. \frac{\partial^2 \mathcal{S}}{\partial x^2} \right|_a = \left. \frac{\partial^2 \mathcal{S}}{\partial x^2} \right|_b = 0.$$

Besides the construction shown here, there are other ways for constructing such a spline, see [4] [5]. A cubic spline  $\mathcal{S}$  can be represented by the formula:

$$\mathcal{S}(x) = a_1 + a_2 x + \frac{1}{12} \sum_{i=0}^n \delta_i |x - x_i|^3. \quad (1)$$

It is a natural spline, if and only if the following conditions hold:

$$\sum_{i=0}^n \delta_i = \sum_{i=0}^n \delta_i x_i = 0. \quad (2)$$

So the natural cubic spline is completely defined by  $\mathbf{a} = (a_1 \ a_2)$  and  $\delta = (\delta_0 \ \delta_1 \ \dots \ \delta_n)$ . Using the definitions

$$\begin{aligned} \mathbf{s}_i &:= \mathcal{S}(x_i) & i = 0, \dots, n, \\ \mathbf{E}_{i,j} &:= \frac{1}{12} |x_i - x_j|^3 & i, j = 0, \dots, n, \\ \mathbf{T}^T &:= \begin{pmatrix} 1 & 1 & \dots & 1 \\ x_0 & x_1 & \dots & x_n \end{pmatrix} \end{aligned}$$

one can express equation (1) as vector-matrix equation:

$$\mathbf{s} = \mathbf{E} \delta + \mathbf{T} \mathbf{a}.$$

The constraints (2) can be written as:

$$\mathbf{T} \delta = 0.$$

Finally, one can find the natural spline interpolant by solving the following equation:

$$\begin{pmatrix} \mathbf{E} & \mathbf{T} \\ \mathbf{T}^T & 0 \end{pmatrix} \begin{pmatrix} \delta \\ \mathbf{a} \end{pmatrix} = \begin{pmatrix} \mathbf{z} \\ 0 \end{pmatrix}. \quad (3)$$

Note, that for the one-dimensional case this is not the fastest algorithm available. It can be found in [5]. One can show, that

$$\int_a^b (\mathcal{S}''(x))^2 dx = \delta^T \mathbf{E} \delta.$$

Finally, the natural cubic spline interpolant  $\mathcal{S}$  minimizes the functional

$$\mathcal{J}(s) = \int_a^b s''(x)dx \quad \forall s \in \mathcal{C}^2((a, b), \mathbb{R}) \quad (4)$$

One can recognize  $\mathcal{J}$  to be a roughness penalty, since  $s''$  is a measure for the curvature of  $s$ . Hence, the natural cubic spline  $\mathcal{S}$  is the twice continuously differentiable function interpolating the data  $(x_i, z_i)_{i=0, \dots, n}$  with the lowest curvature.

## 2.2 Smoothing Spline — one-dimensional

As already mentioned, measurement data consist of statistical noise and a systematic fingerprint. Since any interpolant will pass through all data points, a method solely based on interpolation will not be able to distinguish between these two parts. The concept of interpolating splines is extended by smoothing splines, where the demand  $\mathcal{S}(x_i) = z_i$  is modified, by adding to the roughness penalty (4) the quadratic mean of the distances between spline and data points.

Define for all  $s \in \mathcal{C}^2((a, b), \mathbb{R})$  and  $\lambda \geq 0$ :

$$\mathcal{J}_\lambda(s) := \sum_{i=0}^n (z_i - s(x_i))^2 + \lambda \int_a^b s''(x)dx$$

Starting from the optimality property (4) of the interpolating spline, a smoothing spline  $\mathcal{S}_\lambda$  is defined as the solution of the minimization problem:

$$\mathcal{S}_\lambda = \arg \min_{s \in \mathcal{C}^2((a, b), \mathbb{R})} \mathcal{J}_\lambda(s). \quad (5)$$

The operator  $\arg \min(\mathcal{F}(\cdot))$  returns the argument of  $\mathcal{F}$  which minimizes the value of  $\mathcal{F}$ . The functional  $\mathcal{J}_\lambda(s)$  can be discretized, and one obtains

$$\begin{aligned} \mathcal{J}_\lambda(s) &= (\mathbf{z} - \mathbf{E}\delta - \mathbf{T}\mathbf{a})^T \cdot (\mathbf{z} - \mathbf{E}\delta - \mathbf{T}\mathbf{a}) + \lambda \delta^T \mathbf{E}\delta \\ &= \begin{pmatrix} \delta^T & \mathbf{a}^T \end{pmatrix} \begin{pmatrix} \mathbf{E}^2 + \lambda \mathbf{E} & \mathbf{E}\mathbf{T} \\ \mathbf{T}^T \mathbf{E} & \mathbf{T}^T \mathbf{T} \end{pmatrix} \begin{pmatrix} \delta \\ \mathbf{a} \end{pmatrix} - 2 \begin{pmatrix} \delta^T & \mathbf{a}^T \end{pmatrix} \begin{pmatrix} \mathbf{E} \\ \mathbf{T} \end{pmatrix} \mathbf{z} + \mathbf{z}^T \mathbf{z} \end{aligned}$$

The function  $s$  is defined by  $\delta$  and  $\mathbf{a}$ . If  $\mathcal{J}_\lambda$  is minimal at  $\delta^*$  and  $\mathbf{a}^*$ , the following statements hold:

$$\left. \frac{\partial \mathcal{J}_\lambda(s)}{\partial \begin{pmatrix} \delta \\ \mathbf{a} \end{pmatrix}} \right|_{\begin{pmatrix} \delta \\ \mathbf{a} \end{pmatrix} = \begin{pmatrix} \delta^* \\ \mathbf{a}^* \end{pmatrix}} = 2 \begin{pmatrix} \mathbf{E}^2 + \lambda \mathbf{E} & \mathbf{E}\mathbf{T} \\ \mathbf{T}^T \mathbf{E} & \mathbf{T}^T \mathbf{T} \end{pmatrix} \begin{pmatrix} \delta \\ \mathbf{a} \end{pmatrix} - 2 \begin{pmatrix} \mathbf{E} \\ \mathbf{T} \end{pmatrix} \mathbf{z} \Big|_{\begin{pmatrix} \delta \\ \mathbf{a} \end{pmatrix} = \begin{pmatrix} \delta^* \\ \mathbf{a}^* \end{pmatrix}} \stackrel{!}{=} 0$$

This condition leads to

$$\begin{pmatrix} \mathbf{E}^2 + \lambda \mathbf{E} & \mathbf{E}\mathbf{T} \\ \mathbf{T}^T \mathbf{E} & \mathbf{T}^T \mathbf{T} \end{pmatrix} \begin{pmatrix} \delta^* \\ \mathbf{a}^* \end{pmatrix} = \begin{pmatrix} \mathbf{E} \\ \mathbf{T} \end{pmatrix} \mathbf{z}$$

The following observation allows to simplify this equation:

$$\begin{pmatrix} \mathbf{E} & 0 \\ \mathbf{T}^T & \lambda \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{E} + \lambda \mathbf{I} & \mathbf{T} \\ \mathbf{T}^T & 0 \end{pmatrix} \begin{pmatrix} \delta^* \\ \mathbf{a}^* \end{pmatrix} = \begin{pmatrix} \mathbf{E} & 0 \\ \mathbf{T}^T & \lambda \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{E} & 0 \\ \mathbf{T}^T & \lambda \mathbf{I} \end{pmatrix} \mathbf{z}$$

After a short calculation one obtains, for all  $\lambda > 0$  and  $x_0 < x_1 < \dots < x_n$ :

$$\det \begin{pmatrix} \mathbf{E} & 0 \\ \mathbf{T}^T & \lambda \mathbf{I} \end{pmatrix} \neq 0$$

Hence, it is sufficient to solve the following equation system:

$$\begin{pmatrix} \mathbf{E} + \lambda \mathbf{I} & \mathbf{T} \\ \mathbf{T}^T & 0 \end{pmatrix} \begin{pmatrix} \delta^* \\ \mathbf{a}^* \end{pmatrix} = \begin{pmatrix} \mathbf{z} \\ 0 \end{pmatrix} \quad (6)$$

One can easily see that for  $\lambda \rightarrow 0$  one obtains the interpolating spline. It should be noted, that the way this formula was derived is not valid for  $\lambda = 0$ . Still, one can derive this directly by the observation  $\mathcal{J}_0(s) = \mathcal{J}(s) \quad \forall s \in \mathcal{C}^2((a, b), \mathbb{R})$ .

From  $\lambda \rightarrow \infty$  follows  $\delta^* \rightarrow 0$  and hence the smoothing spline converges to the (linear) regression line. For existence and uniqueness of the smoothing spline the reader is referred to [3].

## 2.3 Smoothing Spline — two-dimensional

The construction of the smoothing and interpolation spline given in the previous section is not the most efficient one in the one-dimensional case, but it can be easily extended to multi-dimensional splines. In this paper only two-dimensional splines are discussed, but the arguments hold for higher dimensions.

Suppose now the data points  $(x_i, y_i, z_i)_{i=0, \dots, n}$  given. And one needs a spline function  $s = s(x, y)$  derived from that data. Again, the spline is defined by its optimality property:

$$\mathcal{S} = \arg \min_{s \in \mathcal{C}^2(\mathbb{R}^2, \mathbb{R})} \mathcal{J}_\lambda(s),$$

with

$$\mathcal{J}_\lambda(s) = \sum_{i=0}^n (z_i - s(x_i, y_i))^2 + \lambda \int_{\mathbb{R}^2} \left( \left( \frac{\partial^2 s}{\partial x^2} \right)^2 + \left( \frac{\partial^2 s}{\partial y^2} \right)^2 + 2 \left( \frac{\partial^2 s}{\partial x \partial y} \right)^2 \right)$$

Define the matrix  $\mathbf{T}$  by

$$\mathbf{T} := \begin{pmatrix} 1 & 1 & \dots & 1 \\ x_0 & x_1 & \dots & x_n \\ y_0 & y_1 & \dots & y_n \end{pmatrix}$$

One can start with an ansatz similar to equations (1) and (2):

$$\begin{aligned} \mathcal{S}(\mathbf{t}) &= \mathbf{T} \mathbf{a} + \sum_{i=0}^n \delta_i \eta(\|\mathbf{t} - \mathbf{t}_i\|_2) \\ \mathbf{T} \delta &= 0 \end{aligned}$$

By defining  $E_{ij} = \eta(\|\mathbf{t}_j - \mathbf{t}_i\|_2)$ , one ends up with the exact same formulation as for the one-dimensional case. The rest of the construction is exactly the same and the properties hold, respectively.

Often,  $\eta(\cdot) : \mathbb{R}^+ \rightarrow \mathbb{R}$  is the basis function. In the one-dimensional case it was sensible to set

$$\eta(r) = \frac{1}{12} |r|^3.$$

In the two-dimensional case the following is often used:

$$\eta(r) = \begin{cases} \frac{1}{16} r^2 \log^{(2)} & r > 0 \\ 0 & r = 0. \end{cases}$$

This spline is referred to as *thin plate spline*.

## 3 Statistical Tests

### 3.1 Introduction to Statistical Tests

This section addresses methods which allow to check if an assumption is satisfied by some data. Such methods are called statistical tests. Since the results are dependent on a random process, those tests might fail. Suppose a given data set and one needs to know if it is Gaussian-distributed or not. Those two statements are called *hypothesis* (Data is Gaussian-distributed) and *alternative* (Data is not Gaussian-distributed). The table 1 shows the four possibilities. In many cases, only one of the two errors can be limited. Hence, one chooses the hypothesis so one can limit the  $\alpha$ -error. Such a test is called *niveau- $\alpha$ -test*, if the probability of an  $\alpha$ -error is equal to  $\alpha$ . This leads to an asymmetry between hypothesis and alternative, so one needs to think about the better-suited formulation. Note that the results of the test are dependent on  $\alpha$ .

Test Result	Real Situation	Hypothesis	Alternative
Hypothesis		result correct	$\beta$ -error
Alternative		$\alpha$ -error	result correct

Table 1: Possibilities for a statistical test

Suppose again the given data set and the hypothesis is that it is Gaussian-distributed. Then, the probability of the niveau- $\alpha$ -test is  $\alpha$  to fail (test result: not normally distributed) and the probability for the test to give the correct result is  $1 - \alpha$ . Often  $\alpha$  is chosen to be 0.05 or 0.01. If the data set does not stem from a Gaussian distribution, one does not know the probability of the test to fail. Of course it is desirable to have a high reliability of the test in this case as well.

Another concept is the  $p$ -value or the level attained: If the  $p$ -value is smaller than an *a priori* chosen limit, one rejects the hypothesis; if the  $p$ -value is bigger, one accepts the hypothesis. More details on statistical tests can be found in [6] and [7].

### 3.2 The D'Agostino-Pearson-Test

In the situation at hand, one needs to decide if the deviations  $dx$  and  $dy$  are normally distributed. The test is based on skewness of a distribution, which is given by

$$g_1^d = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^3}{\sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2}}$$

and the excess

$$g_2^d = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^4}{\left(\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2\right)^2} - 3$$

for discrete data and

$$g_1^c = \frac{(E(X - E(X)))^3}{\sqrt{\text{var}(X)^3}},$$

$$g_2^c = \frac{(E(X - E(X))^4)}{(\text{var}(X))^2} - 3,$$

where  $E$  is the value of expectation and  $\text{var}$  the variance of the property  $X$ . The basic idea of the D'Agostino-Pearson-Test is to compare the values  $g_1^d$  and  $g_2^d$  of the data to those of a Gaussian distribution. This test is also referred to as omnibus test. For more details on this test, please check [8]. An implementation in MATLAB<sup>®</sup> is also available, see [9].

There are alternative tests, like the Shapiro-Wilk test. Two dimensional data, like registration or overlay, can be checked via the following approach:

- Check each dimension separately on normality
- Check for correlation between the two dimensions.

### 3.3 Remarks on normality tests

One needs to be careful when using tests for normality:

Samples with few data points usually pass a normality test. Normality tests have only little power to tell whether a small sample of data comes from a Gaussian distribution or not.

A Q-Q-plot also provides an idea of whether the data stems from a Gaussian distribution.

## 4 Grid Correction

### 4.1 Introduction

Besides changing the design itself, another possibility to change the position of a feature is provided by re-calibrating the tool writing the features on the mask, the so-called pattern generator or writer, by modifying (or correcting) the grid on which it writes. One can envisage two different ways of applying such grid corrections:

- **Polynomial Correction**

The deviations  $dx$ ,  $dy$  are recognized as functions of the position on the mask

$$dx = dx(x, y) \quad (7)$$

$$dy = dy(x, y) \quad (8)$$

Then a polynomial up to  $n^{\text{th}}$  order is removed (residuals of a 3<sup>rd</sup> order fit shown as an example):

$$dx_{\text{res}} = dx - (1 \ x \ y \ x^2 \ xy \ x^2 \ x^3 \ x^2y \ xy^2 \ y^3) \mathbf{a} \quad (9)$$

$$dy_{\text{res}} = dy - (1 \ x \ y \ x^2 \ xy \ x^2 \ x^3 \ x^2y \ xy^2 \ y^3) \mathbf{b}, \quad (10)$$

where  $\mathbf{a}$  and  $\mathbf{b}$  are the coefficients of the polynomial. This method can only correct for slowly varying errors, like the bow of a mask.

- **Mapping Correction**

Since fast-varying distortions cannot be properly corrected by a polynomial approach, an additional method can be used, which works more locally. The residuals of registration data after a polynomial fit on a regular rectangular grid can be used as a template for correction in the following manner: Suppose a point  $(x, y)$  on the mask's frontside lies between four points belonging to the regular grid, such that  $x_l \leq x \leq x_r$  and  $y_l \leq y \leq y_u$ , as shown in figure 1. At the grid points the deviations  $dx$  and  $dy$  are known from the residuals, so one can estimate by bilinear interpolation the deviations at the point  $(x, y)$ :

$$\begin{aligned} dx_{\text{map}} &= (x - x_l)(y - y_l)dx_{lu} + (x - x_r)(y - y_l)dx_{rl} + (x - x_l)(y - y_u)dx_{lu} + (x - x_r)(y - y_l)dx_{ru} \\ dy_{\text{map}} &= (x - x_l)(y - y_l)dy_{lu} + (x - x_r)(y - y_l)dy_{rl} + (x - x_l)(y - y_u)dy_{lu} + (x - x_r)(y - y_l)dy_{ru} \end{aligned}$$

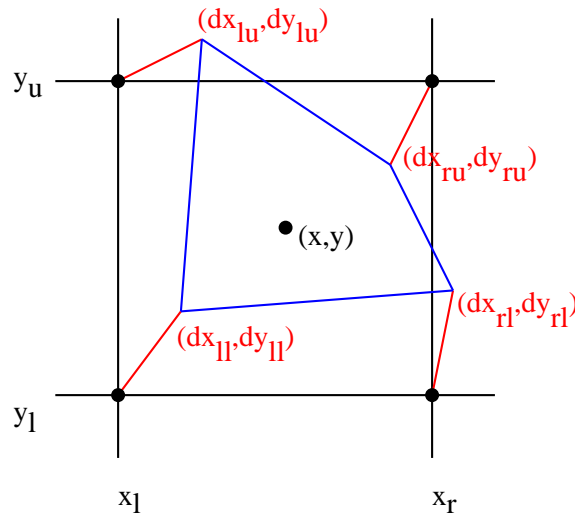


Figure 1: Grid for mapping correction

The basic idea of a grid correction for the improvement of mask's registration is the following:

1. Take registration data of a previously written similar mask,
2. Interpolate the registration data, which is usually only available on a non-regular grid, to a regular grid
3. Apply a polynomial fit to the interpolated data and store the coefficients, as well as the residuals of the polynomial fit (used for mapping correction)
4. Apply to the mask writer's grid the polynomial correction with the stored coefficients and the mapping correction when writing a subsequent mask

## 4.2 Finding the optimal $\sigma$

Since registration data of product masks is only available on non-regular grids, but data on a regular grid is needed for grid correction, one needs to interpolate the data appropriately. Again the choice of the smoothing parameter  $\lambda$  needs to be discussed, since it influences the shape of the interpolant dramatically. Suppose the systematic fingerprint and the number of measurement points of source and destination data are the same. Also, the statistical noise should not change its amplitude or distribution. Then the optimal value for  $\lambda$  is given by the  $\lambda$  that minimizes the registration error of the initial data. On the other hand, one should prefer larger values of  $\lambda$ , since with small values one might 'correct' also a part of the statistical noise, which might induce a rapid worsening of the registration. Corrections with large values of  $\lambda$  will change no part of the statistical noise and eventually the systematic part only partially. Hence, the worst case for large values is given by the original data itself. To be on the safe side, one might multiply the optimal smoothing parameter  $\lambda$  by a number  $\xi > 1$ .

## 4.3 First Results

To estimate the improvements that can be obtained by this correction scheme, simulations were performed. This basic algorithm is sketched in figure 2. As already mentioned, the proposed correction

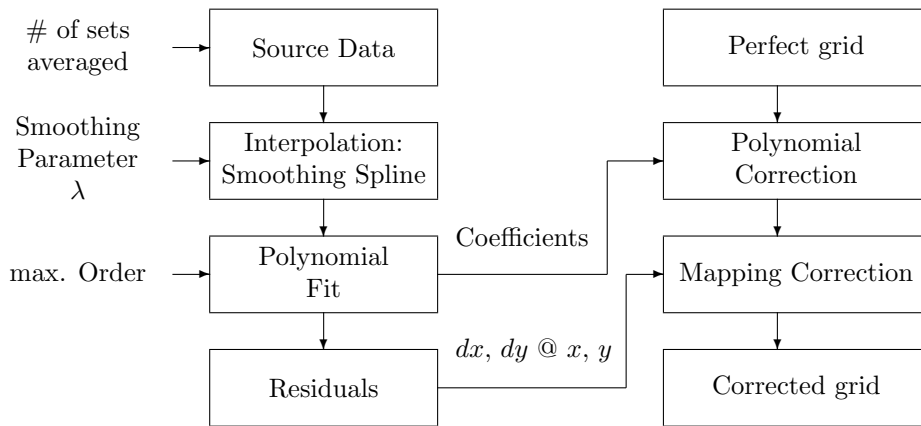


Figure 2: Flow chart of grid correction simulation

scheme is sensitive to any changes in the systematic fingerprint. So data sets are best suited where writing tools and layouts were the same. Masks that needed to be re-written due to issues like rejections for defects or critical dimension errors, which do not influence the registration behaviour, fulfill all those conditions.

Some of those sets were analyzed and the following results were found:

- In virtually every case registration behaviour can be improved by two to three nanometres
- The grid correction is stable with respect to number of grid points and maximal order of polynomial correction
- Choice of the data sets used for deriving the correction and the smoothing parameter  $\sigma$  are crucial to the efficiency of the correction

Results for such a family of masks are shown in table 2. In this study, a grid correction was generated from the registration data of mask 1 and applied to each mask of the family. In this example, too, registration improved for all masks. Finally, a study addressing the stability with respect to the smoothness parameter  $\sigma$  was performed with this data. The results can be seen in figure 3: The optimal parameter according to the method proposed in section 4.2 is around  $\lambda \approx 5000$ . The curves of masks two to four are minimal for higher values of  $\lambda$ . Still, the estimated value seems to be a good choice.

Destination	Source	no correction	1 Set	2 Sets	3 Sets
Mask 1	best (dx/dy)	9.0/10.5	6.5/6.9	6.9/7.3	7.2/7.8
	worst (dx/dy)	—	8.5/8.5	7.6/8.0	6.9/7.4
Mask 2	best (dx/dy)	11.0/10.0	7.2/6.8	7.4/7.2	7.6/7.4
	worst (dx/dy)	—	10.1/9.4	8.9/8.6	8.5/8.3
Mask 3	best (dx/dy)	8.7/12.1	5.9/7.4	7.0/7.9	7.2/8.2
	worst (dx/dy)	—	10.1/10.4	9.2/9.6	8.8/9.2
Mask 4	best (dx/dy)	9.5/11.3	6.6/6.6	7.1/6.8	7.7/7.3
	worst (dx/dy)	—	8.7/8.5	8.2/7.9	7.2/7.0

Table 2: Uncorrected registration and results of grid corrections based on up to three masks (source), applied to four masks (destination). 3 sigma values  $dx/dy$  are given in *nm*.



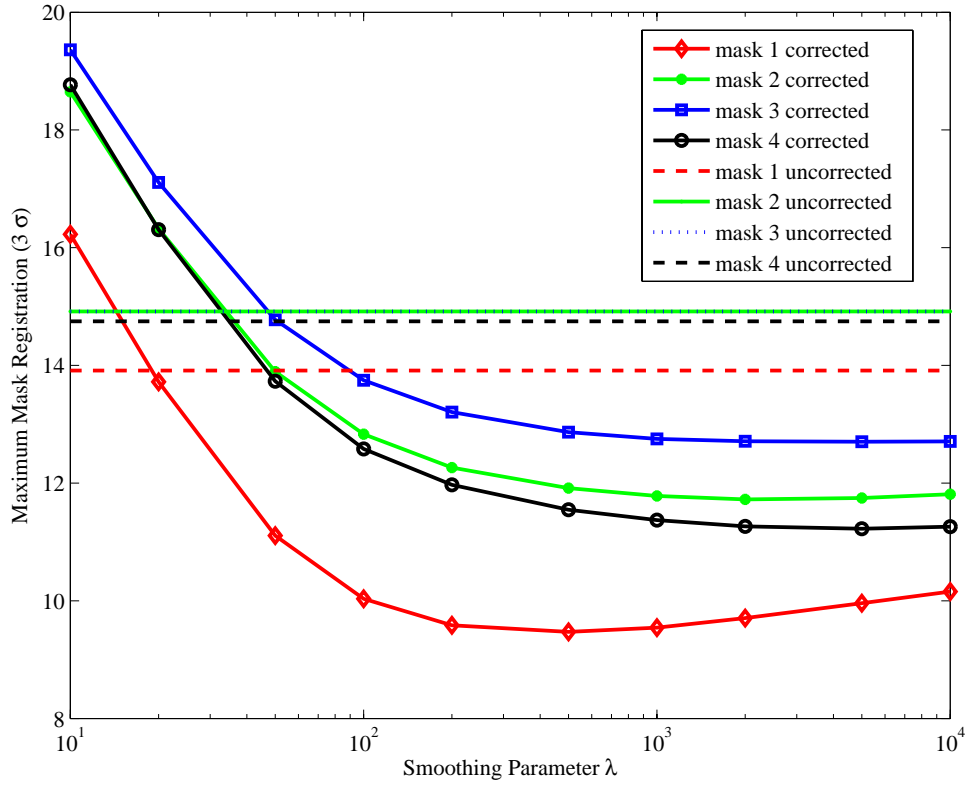


Figure 3: Graph showing the stability of the grid correction applied to four different masks.

## 5 Conclusions

Smoothing splines provide methods for extracting a systematic behaviour from (registration) data, if the systematic fingerprint is smooth. Fast varying signatures are considered as statistical noise. Statistical tests, like the proposed omnibus test, allow an estimation of how well the separation of the systematic fingerprint and the statistical noise works.

This sophisticated data evaluation method allows to verify models or might give rise to certain strategies for optimization, like grid correction. By using the proposed grid correction, one can improve mask registration by up to 3nm. But here it is important to make sure that the systematic fingerprint of the source data does not differ significantly from the data to be corrected. This assumption holds when rewriting a mask with exact the same tools. In this case, the proposed choice of the smoothing parameter was proven empirically to be sensible.

## References

- [1] International Technology Roadmap for Semiconductors. <http://www.itrs.net/>, 2004.
- [2] H.Levinson. *Principles of Lithography*. SPIE Press, second edition, 2005.
- [3] B.W. Silverman P.J. Green. *Nonparametric Regression and Generalized Linear Models*. Chapman & Hall/CPC, first edition, 2000.
- [4] Robert Plato. *Numerische Mathematik Kompakt*. Vieweg, second edition, 2004.
- [5] A. Hohmann P. Deuffhard. *Numerische Mathematik 1, Eine algorithmische Einfhrung*. DeGruyter, second edition, 2002.
- [6] J. Hartung. *Statistik*. Oldenbourg, 10. edition, 1995.
- [7] F. Baur G. Bamberg. *Statistik*. Oldenbourg, 11. edition, 2001.
- [8] J.H. Zar. *Biostatistical Analysis*. Prentice-Hall, second edition, 1999.
- [9] R. Hernandez-Walls A. Trujillo-Ortiz. D'Agostino-Pearon's K2 test assessing normality of data using skewness and kurtosis. a matlab file. <http://www.mathworks.com/matlabcentral/fileexchange/>, 2003.

## Acknowledgements

The authors would like to thank Clemens Utzny (AMTC), Timo Wandel (AMTC), Irmgard Eder (TU Munich) and Carolyn Kalender (TU Munich).

AMTC is a joint venture of AMD, Infineon and Toppan. The AMTC gratefully acknowledges the financial support by the German Federal Ministry of Education and Research (BMBF) under Contract No. 01M3154A (Abbildungsmethodiken fuer nanoelektronische Bauelemente).