Highlights

Improvements to the linear transform technique for generating randomly rough surfaces with symmetrical autocorrelation functions

Michael Watson,Roger Lewis,Tom Slatter

- The linear transform technique of rough surface generation is improved to ensure the resulting ACF is symmetrical
- This guarantees the ACF matches the target ACF in regions which have previously been unfitted
- This removes unwanted high frequency components in line with the coordinate axes which are present with other methods
- Code examples are provided

Improvements to the linear transform technique for generating randomly rough surfaces with symmetrical autocorrelation functions

Michael Watson[a,*], Roger Lewis[a] and Tom Slatter[a]

[a] University of Sheffield, UK

ARTICLE INFO

Keywords:
Linear transformation
Rough surface
Modelling
Surface generation

ABSTRACT

Simulating surfaces with defined surface roughness parameters is a common task in tribology. This task is likely to become more relevant as modelling rough surface contact becomes less costly as it enables parametric studies linking behaviour to roughness parameters independent of a particular surface profile. The linear transform method allows the specification of the autocorrelation function (ACF) and gives some control over the height function, however these methods only fit the linear transformation matrix to half of the ACF. Behaviour outside of this half is unspecified and can lead to large errors.

In this work we show that this problem can be overcome by using a symmetrical linear transformation matrix. This ensures that the resulting ACF is symmetrical. The method given by Liao et al. [1] is extended to include this constraint, including the analytical gradient formula. Additionally, an improvement allowing for the generation of periodic surfaces is given.

The use of a symmetric filter reduced errors in the unfitted region of the ACF, to the same levels as within the fitted region, in one example, this was a reduction from 50% to 3% error. The surface realisations produced by this technique show fewer unphysical effects than those produced by nonsymmetric filters. Particularly, high frequency noise in line with the coordinate axes is removed.

1. Introduction

Simulating realistic surface profiles with defined roughness parameters has long been of interest in tribology. These simulated surfaces can be used in contact models to investigate how behaviour changes with specific aspects of surface roughness or to generalise results beyond a specific surface profile.

Three main methods exist for generating realistic randomly rough surface profiles these are fractal methods, fast Fourier transform (FFT) methods and filter based methods. Fractal methods, such as those based on Hurst or Weierstrass-Mandelbrot functions [2, 3], generate profiles which are self similar up to a maximum scale [4, 5]. These functions allow a broad variety of surface profiles to be generated [6] and make analytical solutions to rough surface contact mechanics problems possible [7, 5]. These methods give complete control over the frequency components in the surface but often lead to unphysical height probability density functions. Additionally, many real surfaces show substantial deviations from self similarity at scales large enough to influence the contact mechanics problem [8]. [1]

FFT methods aim to find the magnitude of the FFT of the surface profile by relating this directly to the power spectral density or auto correlation function. The phase angles of the FFT are then randomly set [10, 11] and the inverse FFT taken to give a surface profile. In general, these methods give no control over the height probability distribution, however, a recent iterative technique presented by Perez et al. [12] allows this to be specified exactly, unfortunately the convergence criteria for this technique are not known. In addition, large errors are often present between the actual ACF and the intended ACF, especially when long autocorrelation lengths are used [11].

Filter based methods aim to generate a surface profile by filtering an array of uncorrelated random values, such as those that can easily be generated by a random number generator [13, 14]. The result is a surface with the desired auto correlation function, with extensions allowing the skew and kurtosis of the height probability distributions to be set [15]. The filter coefficients can be found either by optimisation [16, 1, 14] or analytically through the FFT properties

*Corresponding author
mike.watson@sheffield.ac.uk (M. Watson)
ORCID(s): 0000-0003-1256-8755 (M. Watson)

[1] A good numerical implementation of the Hurst fractal method in presented by [9]

of the autocorrelation function [13], however analytical solutions are often poor, especially for long autocorrelation lengths [11].

The main problems with optimising for the filter coefficients are speed of computation and the introduction of unwanted artefacts, typically high frequency components in line with the coordinate axes. These artefacts are not present with other surface generation techniques or when analytical solutions are used for the filter coefficients. Since the optimisation problem was first presented by Patir [14], there have been many advances in computing power and efficiency improvements to the problem [10, 1]. These have made fitting larger filters feasible, hence allowing surfaces with large autocorrelation lengths to be generated accurately. However, the unwanted artefacts are rarely discussed, though they are often clearly present [10].

In this work we will show that these artefacts are due to problems in the original optimisation problem presented by Patir [14]. Specifically, the problem in its original form allows only one quarter of the target ACF to be fitted. While compliance in the opposite quarter of the function is guaranteed due to rotational symmetry of the ACF ($R(x, y) \equiv R(-x, -y)$), the remaining half of the function is not fitted ($R(x, y) \neq R(x, -y)$). This undefined behaviour is never desirable as it leads to large errors in the overall ACF of the surface. In this work we have shown that by enforcing a symmetry constraint on the filter coefficients, behaviour over the entire ACF can be specified, if the ACF has reflectional symmetry, which is often specified. These improvements allow researchers to use filter based methods to accurately model surfaces with long autocorrelation lengths and non Gaussian height distributions.

2. Background

A random rough surface (z) of size $[N, M]$ can be generated through convolution between a $[N+n, M+m]$ matrix of uncorrelated Gaussian random numbers (η) and a $[n, m]$ matrix of filter coefficients (α) [14] [2]:

$$z_{i,j} = \sum_{k=1}^{n} \sum_{l=1}^{m} \alpha_{k,l} \eta_{i+k,j+l} \tag{1}$$

As the values in η are not correlated, η can be easily generated by a random number generator. The ACF of a surface is expressed, in discrete terms as [14]:

$$\begin{aligned} R_{p,q} &= \frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} z_{i,j} z_{i+p,j+q} \\ &= E[z_{i,j} z_{i+p,j+q}] \end{aligned} \tag{2}$$

where $E[]$ is the expectation or averaging operator. As the values in η are independent and have unit variance the expectation of any pair of values is zero while the expectation of a single value squared is 1. Patir [14] showed that this allows the expectation of the ACF of the surface ($E[R^{zz}_{p,q}]$) to be written in terms of the filter coefficients ($\alpha_{k,l}$) only:

$$\begin{aligned} E\left[R^{zz}_{p,q}\right] &= \sum_{k=1}^{n-p} \sum_{l=1}^{m-q} \alpha_{k,l} \alpha_{k+p,l+q} \\ &= \alpha \circledast \alpha_f; \quad p = 0...n,\ q = 0...m \end{aligned} \tag{3}$$

where α_f is α flipped in both dimensions and subsequently padded with n-1 and m-1 zeros in the i and j directions respectively. This form allows this step to be accelerated by FFT convolution [1].

This is a set of simultaneous non linear equations which can be solved for the filter coefficients to give the desired ACF. In the original work by Patir, the above equation is applied over the range $p = 0, 1, .., n-1,\ q = 0, 1, ..., m-1$, however in reality p and q are not restricted to be non-negative. The result of this is that behaviour of the ACF is only specified in the quadrant: $p \geq 0, g \geq 0$ and, by rotational symmetry of the ACF: $p \leq 0, q \leq 0$. An example full ACF is shown in Figure 1A, with the fitted and unfitted quadrants indicated in Figure 1B. To reiterate, the method presented

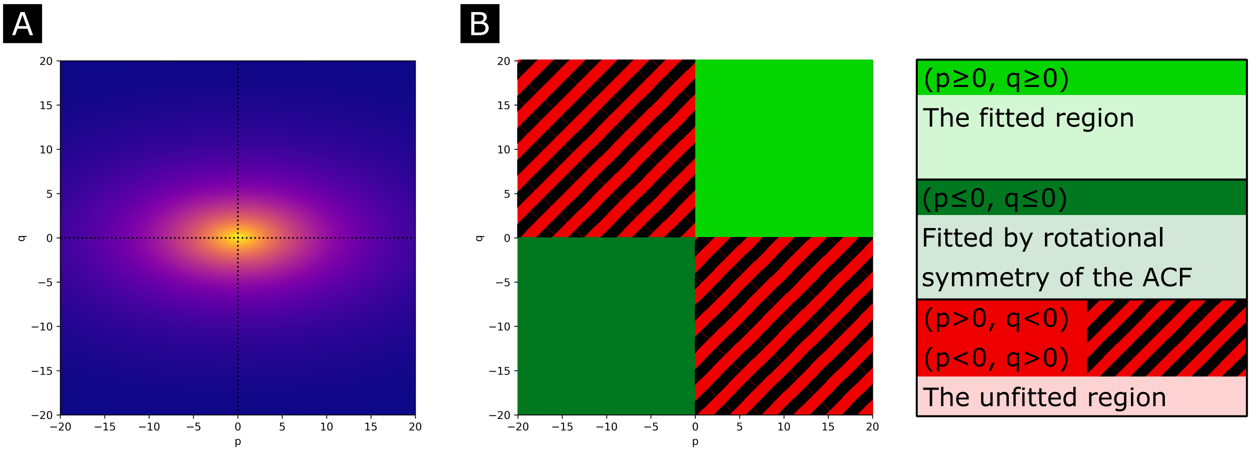

Figure 1: A and B showing an example full ACF (A) and a diagram indicating the regions of the ACF which are fitted by the method presented by Patir (B)

by Patir, and all improvements to this method, generate surfaces which fit the target ACF only in the fitted regions indicated in Figure 1B, behaviour in the unfitted regions is undefined.

Practically, in generated surfaces, it is almost always desirable to have reflectional symmetry of the ACF ($R_{p,q} = R_{-p,q}$). It is clear from Equation 3 that this is enforced if the filter coefficients matrix has reflectional symmetry about $\alpha_{0,j}$ or $\alpha_{i,0}$. As 2D ACFs must be rotationally symmetric about the origin ($R_{p,q} = R_{-p,-q}$), enforcing reflectional symmetry in either $p = 0$ or $q = 0$ is sufficient to achieve reflectional symmetry in both ($R_{p,q} = R_{-p,q} = R_{p,-q} = R_{-p,-q}$).

In this work, the symmetry constraint will be applied to the formulation of the problem given by [1]. However, the problems described and solved in this work apply to any method which solves the problem described by Patir. In this formulation, the problem is transformed to the non linear least squares problem shown below. This greatly improves the computation speed without altering the underlying problem.

$$\begin{aligned} min\ f(\alpha_{k,l}) &= \sum_{p=0}^{n-1}\sum_{q=0}^{m-1}\left(R^{target}_{p,q} - E\left[R^{zz}_{p,q}\right]\right)^2 \\ g(\alpha_{k,l}) = \nabla f(\alpha_{k,l}) &= \left(\frac{\partial f}{\partial \alpha_{1,1}}, \frac{\partial f}{\partial \alpha_{1,2}}, \ldots, \frac{\partial f}{\partial \alpha_{n,m}}\right) \end{aligned} \tag{4}$$

where $R^{target}_{p,q}$ is the target ACF to be fitted, $E\left[R^{zz}_{p,q}\right]$ is the expectation of the ACF, given by Equation 3, $f(\alpha)$ is the objective function and $g(\alpha)$ is the gradient of the objective function with respect to each of the filter coefficients given by [1]:

$$g(\alpha_{i,j}) = \sum_{p=0}^{n-i}\sum_{q=0}^{m-j} 2\left(R^{target}_{p,q} - E\left[R^{zz}_{p,q}\right]\right)(-\alpha_{i+p,j+q}) + \sum_{p=0}^{i-1}\sum_{q=0}^{j-1} 2\left(R^{target}_{p,q} - E\left[R^{zz}_{p,q}\right]\right)(-\alpha_{i-p,j-q}) \tag{5}$$

[2] Alternatively, if a periodic surface is required η can be a $[N, M]$ matrix or unique random numbers, padded to $[N + n, M + m]$ by wrapping values. To the authors knowledge, this adaptation has not previously been published, but it is too trivial for it's own work.

3. Methodology

3.1. Implementation of a symmetric filter

The method described above can be used with very little modification. As the values in η are independent, the location of the filter does not affect the result [3]. This means that the same equations for the objective function and gradients can be used with the full, reflected filter coefficients matrix (α^* size $[2n-1, m]$). The output of this function is simply truncated to the same size as the target ACF $[n, m]$. The objective function can then be calculated as above.

The gradient function can also be used with very little modification. As differences between the expectation of the ACF and the target ACF do not affect the objective function where $p > n$ the limits of the summation can be reduced to save computation time and avoid out of bounds indexing.

$$g^*(\alpha^*_{i,j}) = \sum_{p=0}^{min(a-i,n)} \sum_{q=0}^{m-j} 2\left(R^{target}_{p,q} - E\left[R^{zz}_{p,q}\right]\right)\left(-\alpha^*_{i+p,j+q}\right) + \sum_{p=0}^{min(i-1,n)} \sum_{q=0}^{j-1} 2\left(R^{target}_{p,q} - E\left[R^{zz}_{p,q}\right]\right)\left(-\alpha^*_{i-p,j-q}\right) \quad (6)$$

where $a = 2n-1$. This gives the same number of gradients as filter coefficients in the full, reflected filter coefficients matrix ($(2n-1)m$). These values can be combined by simple addition as shown below:

$$g(\alpha_{i,j}) = \begin{cases} g^*_{n,j}, & \text{if } i = 1, \\ g^*_{n+i-1,j} + g^*_{n-i+1,j}, & \text{otherwise} \end{cases} \quad (7)$$

Computation of both the objective function and gradient matrix is slower with a symmetric filter coefficients matrix. However, the number of variables in the non linear optimisation problem does not increase. A Python implementation of this code, as well as code to generate all figures in this manuscript are supplied in the additional material.

3.2. Comparison between symmetric and nonsymmetric filters

This section describes the methodology used to compare the updated method with the original method. To do this, target auto correlation functions are generated with an exponential function as described by Patir [14]. The equation of this function is given below:

$$R^{target}_{p,q} = \sigma^2 exp\left\{-2.3\left[\left(\frac{p\delta_x}{\tau^*_x}\right)^2 + \left(\frac{q\delta_y}{\tau^*_y}\right)^2\right]^{\frac{1}{2}}\right\} \quad (8)$$

In which σ^2 is the rms roughness of the surface, δ_x and δ_y are the spacing of the grid points in each direction and τ^*_x and τ^*_y are the lengths at which the AFC decays to 10% of it's original value. Below the performance of each technique is evaluated by fitting to three different autocorrelation functions:

- $\tau^*_x/\delta_x = 10$, $\tau^*_y/\delta_y = 10$, $\sigma = 1$
- $\tau^*_x/\delta_x = 10$, $\tau^*_y/\delta_y = 20$, $\sigma = 1$
- $\tau^*_x/\delta_x = 10$, $\tau^*_y/\delta_y = 50$, $\sigma = 1$

These are fitted over three different relative filter sizes: two, four and five times the τ^*/δ value in each dimension. For example the $\tau^*_x/\delta_x = 10$, $\tau^*_y/\delta_y = 50$ case will be tested with filter sizes of 20×100, 40×200 and 50×250. In the case of symmetric filters the filter size refers to the number of unique filter coefficients in each dimension.

Where areas of the ACF are compared the error is given as the norm of the point-wise error divided by the norm of the ACF over the same region as shown in Equation 9. This error metric differs from those presented in previous works. Manesh et al. [15] compared roughness parameters, as these depend only on values from the ACF along the

[3] A filter with indices for i from $-n$ to n which is symmetric about $i = 0$ will give the same expected ACF as one with indices from 1 to $2n+1$, however the values for the profile height z will be different

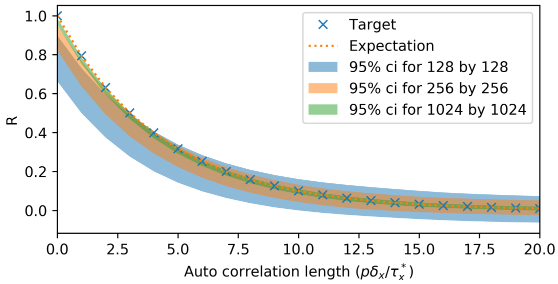

Figure 2: Showing the effect of the size of the realisation on the measured ACF, values for the ACF only taken along the y axis. Example shown is $\tau_x^*/\delta_x = 10$, $\tau_y^*/\delta_y = 10$, $\sigma = 1$, fitted by a 30x30 filter. Shaded regions show 95% confidence intervals, by Monte Carlo simulation.

lines $p = 0$ and $q = 0$ this metric is not suitable for this study. While Liao et al. [1] took the mean of the error values over the fitted quarter of the ACF, the methodology is not clear and this method seems likely to underestimate the true error or be dominated by insignificant changes to small values.

$$error_{region}\left(R^{actual}\right) = \frac{\left(\sum_i \left(R_i^{target} - R_i^{actual}\right)^2\right)^{0.5}}{\left(\sum_i \left(R_i^{target}\right)^2\right)^{0.5}} \quad \text{for i } \in \text{ region} \tag{9}$$

The measured ACF of any particular realisation of a filtered surface is dependent on the actual values in the Gaussian random number sequence (η) and the size of the surface. The expectation of this value, which can also be though of as the ACF of an infinitely large surface, or equally as the true ACF without edge effects, is given by Equation 3. This is illustrated in Figure 2. For this reason, when comparing measured ACFs to the target ACF the average of 10 realisations, each 2048 by 2048 points will be used. However, when plotting surface realisations for display purposes 256 by 256 point realisations will be used to improve visibility of surface features.

4. Results

Example surface realisations generated by the symmetric filter method are shown in Figure 3 A-C. Also shown are the fitted regions of the target ACFs (D-F) and the expected ACFs (G-I) in this region for each of the surfaces for the largest scale of filter. As shown, in the fitted region, an extremely close fit is achieved between the expected ACF and target ACF in each case. Similar results are found for the ‘original’ method presented by [1].

Figures 4 A-C, show the unfitted regions of the ACF measured from example surface realisations generated by the original method (non-symmetric filter). The corresponding ACFs for surfaces generated with a symmetric filter are shown in Figures 4D-F, there are clear differences between the methods in this region of the ACF. The error between the measured and target ACF in this region is shown in Figures 4G-I and Figures 4J-L for the original and symmetric methods respectively. While large discrepancies are present in between the target ACF and ACFs measured from surfaces generated by the original method (Figures 4G-I), these errors are too small to be seen on this scale for the symmetric filter method. This clearly shows the limitations of the method presented by Patir [14], in terms of the unfitted quadrants of the ACF.

Numerical results for the error between the target ACF and the measured ACFs for each of the cases investigated are shown for both the fitted and unfitted region in table 1. Again these clearly show extremely large errors outside the fitted region when nonsymmetric filters are used. These errors are reduced to similar values as those found in the fitted region when symmetric filters are used.

The errors in table 1 have been computed over the same area of ACF which is used to fit the filter in the optimisation step. This area can be expanded to give an indication of the benefit seen from using a larger filter. The error over a 50 by 250 region of the ACF for surfaces made by the 20 by 100 filter ($\tau_y^*/\delta_y = 50$), was only 16%, this is an increase

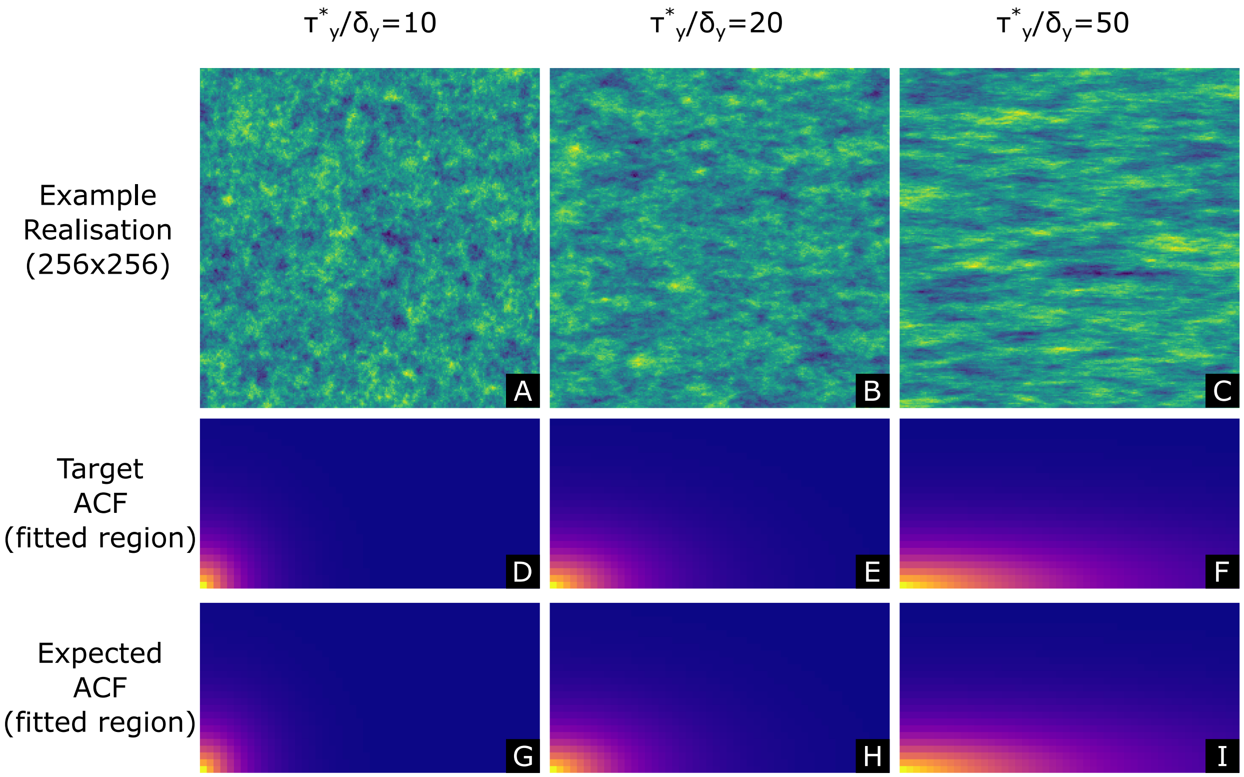

Figure 3: A-C: Example surface realisations each 256 by 256 points generated by the largest scale symmetrical filter, D-F: 25 by 50 point sections of the target ACFs from the fitted region, G-I: 25 by 50 point sections of the expectation of the ACF for each of the filters

$\frac{\tau_y^*}{\delta_y}$	Filter shape	Error - fitted region $p, q \geq 0$		Error - unfitted region $p \leq 0, q \geq 0$	
	(pts)	Original	Symmetric	Original	Symmetric
10	20x20	$1.8 \pm 0.53\%$	$1.5 \pm 0.46\%$	$24.6 \pm 0.41\%$	$1.7 \pm 0.45\%$
	40x40	$3.3 \pm 0.91\%$	$3.4 \pm 0.68\%$	$22.6 \pm 0.73\%$	$3.3 \pm 0.49\%$
	50x50	$4.1 \pm 0.56\%$	$4.4 \pm 0.75\%$	$23.2 \pm 0.74\%$	$4.3 \pm 0.65\%$
20	20x40	$2.3 \pm 1.09\%$	$2.6 \pm 0.50\%$	$19.7 \pm 0.97\%$	$2.9 \pm 0.99\%$
	40x80	$4.2 \pm 0.72\%$	$4.8 \pm 0.58\%$	$24.0 \pm 1.53\%$	$4.7 \pm 0.71\%$
	50x100	$5.4 \pm 0.66\%$	$6.3 \pm 0.95\%$	$23.9 \pm 1.27\%$	$6.8 \pm 1.63\%$
50	20x100	$3.4 \pm 0.97\%$	$3.9 \pm 1.32\%$	$50.9 \pm 1.35\%$	$3.3 \pm 0.85\%$
	40x200	$6.8 \pm 1.24\%$	$7.4 \pm 0.98\%$	$30.9 \pm 1.64\%$	$7.8 \pm 1.84\%$
	50x250	$8.2 \pm 0.50\%$	$10.4 \pm 1.56\%$	$25.7 \pm 1.48\%$	$9.8 \pm 1.64\%$

Table 1
Numerical results for errors from every case in this work, ± indicates the standard deviation across ten 2048 by 2048 point realisations. Errors are computed over an area of the ACF equal to the filter shape.

of 6% over the 50 by 250. For strongly directional ACFs, errors outside the fitted region but in the fitted quadrant, are small compared to errors from outside the fitted quadrant. Thus, increasing the filter size without addressing the errors from outside the fitted quadrant is likely to be a poor method for reducing overall error in the ACF.

These errors are clearly present on the surface realisations for small filters. Figure 5 shows three realisations of surfaces with the same target ACF ($\tau_y^*/\delta_y = 20$). The realisation generated by the 20 by 40 point filter (Figure 5 A) shows the unphysical and undesirable phenomena of high frequency texture aligned with the coordinate axes that is typical with this technique. Subjectively, this texture gives the realisation a 'printed on fabric' appearance. This is still present when a large nonsymmetric filter is used as shown in Figure 5C. However these effects are largely eliminated by a small symmetric filter as shown in Figure 5B and no longer visible when a large symmetric filter is used as shown

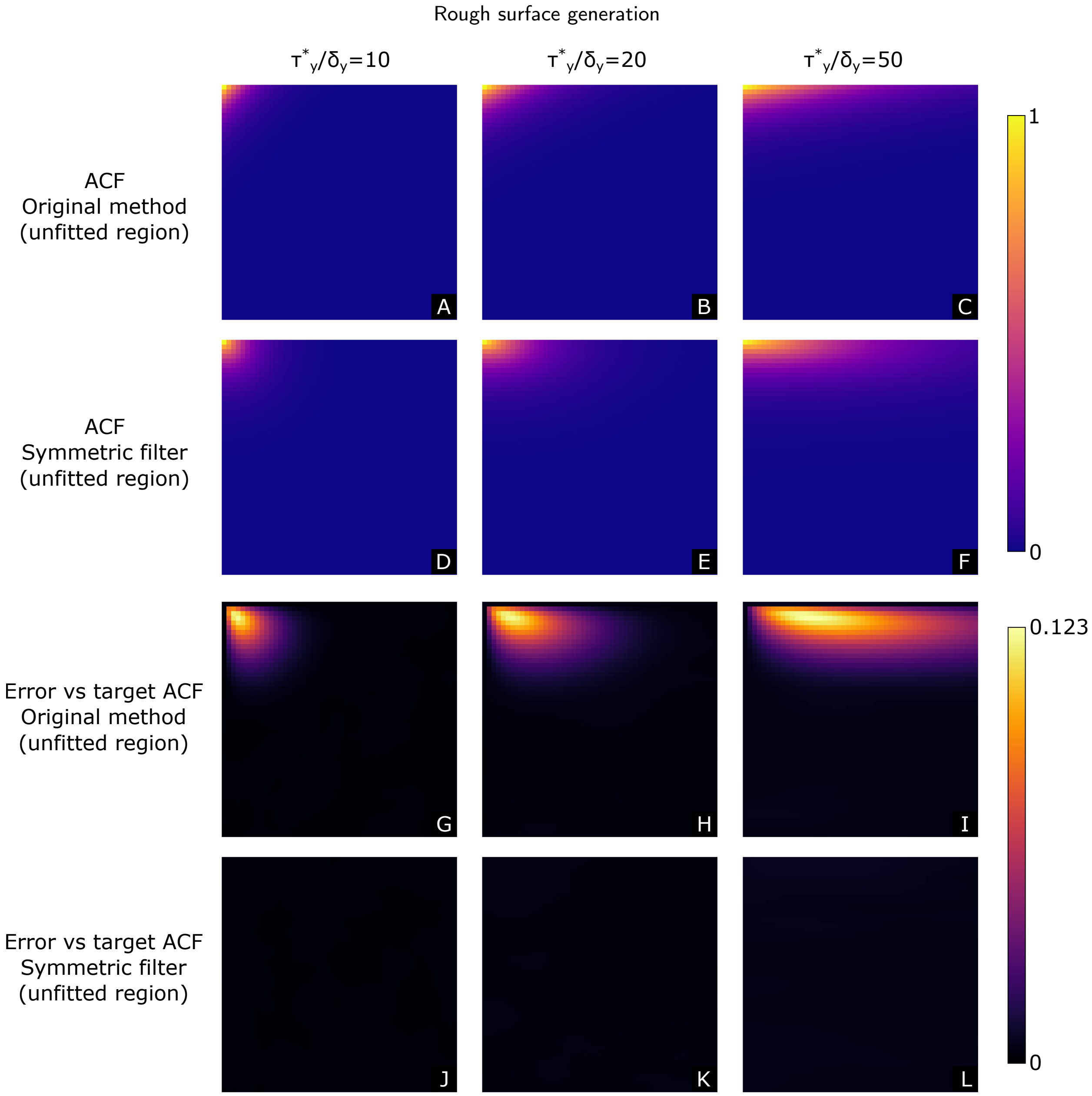

Figure 4: A-F The unfitted region of ACFs measured as an average from 50, 2048 by 2048 point surface realisations generated by the original method (A-C) and the symmetric filter method (D-F). G-L The absolute error between the ACFs shown in figures A-F and the target ACF.

in Figure 5D.

These high frequency components are more clearly shown in the power spectra of the surfaces, Figure 5 E-H. As shown, with the nonsymmetrical filter (Figure 5G) the surface contains unwanted, strong, high frequency components in line with the coordinate axes, these are eliminated when a symmetrical filter is used.

As the power spectral density is the Fourier transform of the ACF this result can be understood by considering that the symmetrical filter removes the sharp change in the ACF at the edge of the fitted region, at the lines $p = 0$ and $q = 0$.

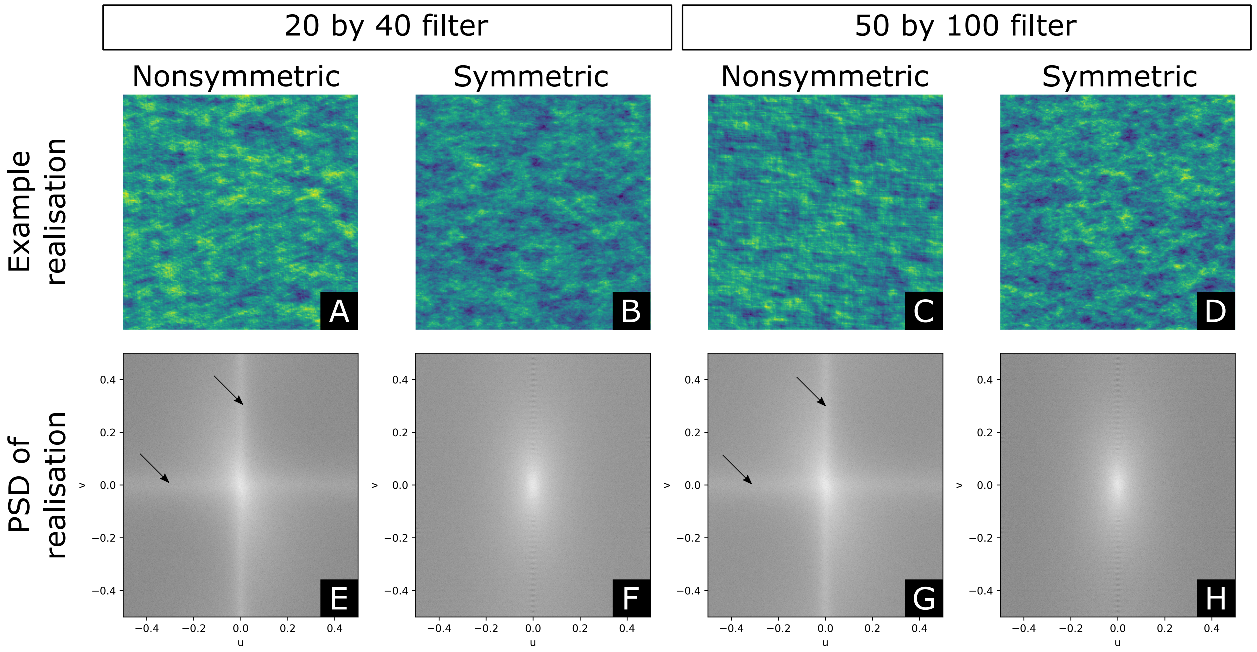

Figure 5: Surface realisations produced by different filters fitted to the same autocorrelation function ($\tau_y^*/\delta_y = 20$) (A-D) and power spectral densities of equivalent realisations (E-H). A: 20 by 40 filter fitted with the original method, B: 20 by 40 symmetrical filter, C: 50 by 100 fitted with the original method and D: symmetrical filter. E-H are power spectral densities of 2048 by 2048 point realisations equivalent to the realisations shown in A-D respectively. Arrows indicate unphysical, strong, high frequency components in line with the coordinate axes

5. Conclusions

The rough surface generation method suggested by Patir, which has been widely adopted and extended by multiple other works, has been shown to produce inaccurate results in important areas of the ACF outside of the fitted region. Errors of up to 50% of the magnitude of the ACF are found in this region leading to unwanted directionality and high frequency components in line with coordinate axes in generated surfaces.

This problem is solved if the filter found in the non linear optimisation step is constrained to be symmetric. With a symmetric filter the expectation of the ACF in the unfitted region is the same as the fitted region. This change to the optimisation problem is compatible with any of the extensions or optimisations to the original method.

Example realisations generated with relatively small symmetric filters show fewer unphysical or unwanted features and no high frequency components in line with the coordinate axes. This shows that errors in the unfitted region of the ACF are important for the generated surfaces and suggests that errors typically attributed to the use of too small a filter can be resolved by adopting this technique more efficiently than simply using a larger filter.

6. Acknowledgements

This work was funded through the EPSRC Programme Grant "Friction: The Tribology Enigma" EP/R001766/1

CRediT authorship contribution statement

Michael Watson: Conceptualization of the study, Methodology, Software, Writing - original draft. **Roger Lewis:** Funding acquisition, Writing - review & editing. **Tom Slatter:** Funding acquisition, Writing - review & editing.

References

1. Liao, D., Shao, W., Tang, J., Li, J.. An improved rough surface modeling method based on linear transformation technique. *Tribology International* 2018;119(August 2017):786–794. URL: `https://doi.org/10.1016/j.triboint.2017.12.008`. doi:`10.1016/j.triboint.`

2017.12.008.

2. Ausloos, M., Berman, D.H.. A mulivariate Weierstrass-Mandelbrot function. *Proc R Soc Lond A* 1985;(400):331–350.
3. Berry, M.V., Lewis, Z.V.. On the Weierstrass-Mandelbrot fractal function. *Proceedings of the Royal Society of London A Mathematical and Physical Sciences* 1980;370:459–484. doi:10.1098/rspa.1980.0044.
4. Persson, B.N.J.. On the Fractal Dimension of Rough Surfaces. *Tribology Letters* 2014;54(1):99–106. URL: http://link.springer.com/10.1007/s11249-014-0313-4. doi:10.1007/s11249-014-0313-4.
5. Yastrebov, V.A., Anciaux, G., Molinari, J.F.. The role of the roughness spectral breadth in elastic contact of rough surfaces. *Journal of the Mechanics and Physics of Solids* 2017;107:469–493. URL: http://dx.doi.org/10.1016/j.jmps.2017.07.016. doi:10.1016/j.jmps.2017.07.016.
6. Thielen, S., Magyar, B., Piros, A.. Reconstruction of three-dimensional turned shaft surfaces with fractal functions. *Tribology International* 2016;95:349–357. URL: http://dx.doi.org/10.1016/j.triboint.2015.11.028. doi:10.1016/j.triboint.2015.11.028.
7. Majumdar, A., Bhushan, B.. Role of fractal geometry in roughness characterization and contact mechanics of surfaces. *Journal of Tribology* 1990;112(2):205–216. doi:10.1115/1.2920243.
8. Campañá, C., Müser, M.H.. Contact mechanics of real vs. randomly rough surfaces: A Green's function molecular dynamics study. *Epl* 2007;77(3). doi:10.1209/0295-5075/77/38005.
9. Kanafi, M.M.. Surface generator: artificial randomly rough surfaces. 2020. URL: https://www.mathworks.com/matlabcentral/fileexchange/60817-surface-generator-artificial-randomly-rough-surfaces.
10. Francisco, A., Brunetière, N.. A hybrid method for fast and efficient rough surface generation. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology* 2016;230(7):747–768. doi:10.1177/1350650115612116.
11. Wu, J.J.. Simulation of rough surfaces with FFT. *Tribology International* 2000;33(1):47–58. doi:10.1016/S0301-679X(00)00016-5.
12. Pérez-Ràfols, F., Almqvist, A.. Generating randomly rough surfaces with given height probability distribution and power spectrum. *Tribology International* 2019;131(October 2018):591–604. URL: https://doi.org/10.1016/j.triboint.2018.11.020. doi:10.1016/j.triboint.2018.11.020.
13. Hu, Y., Tonder, K.. Simulation of 3-D random rough surface by 2-D digital filter and Fourier analysis. *International Journal of Machine Tools and …* 1992;32(1-2):83–90. URL: http://www.sciencedirect.com/science/article/B6V4B-485Y6HN-GJ/2/d82ae0f3a2b3dc5b39cdee1222b85e16{%}5Cnhttp://www.sciencedirect.com/science?{_}ob=ArticleURL{&}{_}udi=B6V4B-485Y6HN-GJ{&}{_}user=6691966{&}{_}coverDate=04/30/1992{&}{_}rdoc=1{&}{_}fmt=high{&}{_}orig=search{&}{_}sort=d{&}{_}doca. doi:10.1016/0890-6955(92)90064-N.
14. Patir, N.. A numerical procedure for random generation of rough surfaces. *Wear* 1978;47(2):263–277. doi:10.1016/0043-1648(78)90157-6.
15. Manesh, K.K., Ramamoorthy, B., Singaperumal, M.. Numerical generation of anisotropic 3D non-Gaussian engineering surfaces with specified 3D surface roughness parameters. *Wear* 2010;268(11-12):1371–1379. URL: http://dx.doi.org/10.1016/j.wear.2010.02.005. doi:10.1016/j.wear.2010.02.005.
16. Liao, D., Shao, W., Tang, J., Li, J., Tao, X.. Numerical generation of grinding wheel surfaces based on time series method. *The International Journal of Advanced Manufacturing Technology* 2018;94(1-4):561–569. URL: http://link.springer.com/10.1007/s00170-017-0868-y. doi:10.1007/s00170-017-0868-y.