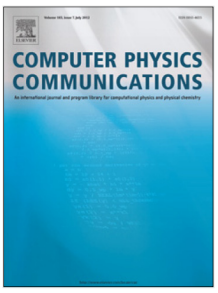
Journal Pre-proof



A new mathematical interpretation of disordered nanoscale material systems for computational modelling

Jack J. Tyson, Tasmiat Rahman, Stuart A. Boden

PII: S0010-4655(20)30183-1
DOI: https://doi.org/10.1016/j.cpc.2020.107399
Reference: COMPHY 107399

To appear in: *Computer Physics Communications*

Received date : 28 October 2019
Revised date : 7 May 2020
Accepted date : 24 May 2020

Please cite this article as: J.J. Tyson, T. Rahman and S.A. Boden, A new mathematical interpretation of disordered nanoscale material systems for computational modelling, *Computer Physics Communications* (2020), doi: https://doi.org/10.1016/j.cpc.2020.107399.

Updated Manuscript (Changes Highlighted)

A New Mathematical Interpretation of Disordered Nanoscale Material Systems for Computational Modelling

Jack J. Tyson*, Tasmiat Rahman and Stuart A. Boden

School of Electronics and Computer Science, University of Southampton, SO17 1BJ, UK

*Corresponding author.

Email addresses: j.j.tyson@southampton.ac.uk (J.J. Tyson), t.rahman@soton.ac.uk (T. Rahman), sb1@ecs.soton.ac.uk (S.A. Boden)

Abstract

As the era of microscale technologies becomes increasingly overcome by that of the nanoscale, an ever-increasing emphasis on the accurate modelling of such scaled systems is apparent. This work explores the combination of the finite element method with a new set of statistical algorithms to model the optical properties of disordered nanoscale morphologies. A silicon surface textured with a random distribution of nanowires is created to simulate, as an example study, how it responds to incident light. By averaging over many iterations of the model in which the structural parameters are varied around average values, a good match to experiment is achieved, showcasing an error as low as 1.34% in magnitude against measured data. This research introduces a fresh computational approach to simulating heterogeneous material structures widely applicable for modelling across the field of nanotechnology.

Keywords: *materials, modelling, nanoscale, COMSOL, heterogeneous*

1) Introduction

Modern computational and mathematical methods permit us to recreate many different physical conditions to model various systems accurately. The finite modelling methods are at the forefront of this area with unparalleled accuracy, robustness and versatility. In particular, the finite element method (FEM) has been hailed as one of the most powerful tools in electromagnetic (EM) wave optics [1,2] due to its unmatched level of control over its simulation domains, yielded in-part due to its adaptive free-tetrahedral approach to meshing. The FEM has also been showcased to work well for fluid dynamics, thermal propagation and structural mechanics [3]. Modelling such problems using this method heralded an interest in the late 1990s, whereby increasingly more complex structures needed to be modelled, and often in-place of their fabrication [3]. This same level of interest has lingered for the modelling of structures with further complexity at even smaller scales and remained the principal motivation of the work reported here [4].

Vertical, protruding silicon nanowires (SiNWs) are of interest within the field of nanotechnology as a result of their broad set of applications. This is inclusive of areas within optics [5-7], nanosensors [8], nanoelectronics [9] and photovoltaics [10-13]. Such structures, regardless of their application, are commonly formed through one of two “top down” fabrication methods: reactive ion etching and metal-assisted chemical etching. Despite each type of etch producing slightly differing SiNW properties, surface heterogeneity is constantly observed across all of them. Variations between individual SiNWs, as well as their tendency to bunch together unpredictably, are visibly apparent in many distinct studies [10,11,13-18]. Attempts have been made, prior to this work, to model SiNWs with largely varying accuracy as a result of these unpredictable properties. These attempts include using the effective index (EI) [19], finite difference time domain (FDTD) [20], and finite element [21,22] methods. Given that all computational methods rely on periodicity to a lesser or greater degree, truly heterogeneous structures like SiNWs are impossible to simulate with complete accuracy. Instead, we take a fresh mathematical and iterative approach to simulating such material systems, used in conjunction with the FEM and COMSOL Multiphysics®†, to give a model with a higher degree of reliability and accuracy.

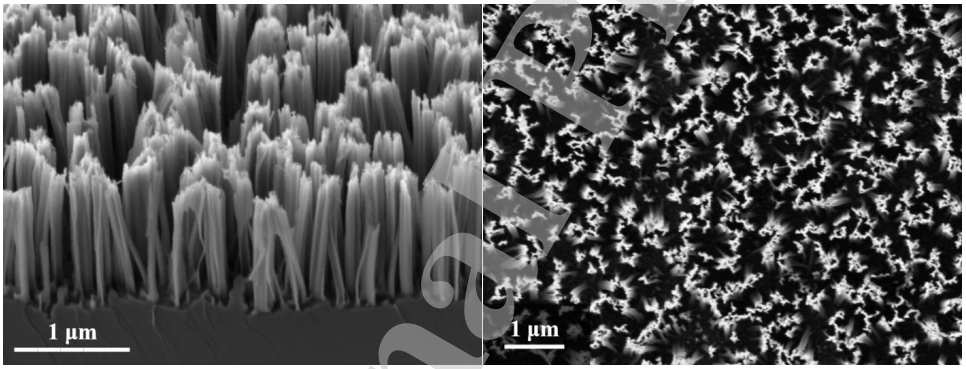
2) Modelling Methods

Effective pseudo-randomisation requires a high degree of sequential mathematical processing, to which a substantial number of *'random'* functions, logical switches and iterative calculations must be employed before any geometry is even realised. The entire model reported here was reduced to be inclusive of three key pre-processing stages, listed respective of their execution order:

1. Randomisation and constraint;
2. Generation;
3. Meshing.

Post-processing was limited to a single step process, whereby the number of geometry iterations, n, required for convergence could be distributed appropriately for the system architecture. On multi-parallel computer systems, a network consisting of multiple individually addressable multi-thread compute nodes, this can be done immediately given that the point of convergence n_c is already known. For serial or low-powered computer systems, this would run in the form of a closed loop; one iteration after another, at the cost of substantial runtime. The point of data convergence is a crucial study that must be carried out prior to simulation on parallel-execution systems for a given simulation geometry to be modelled accurately and efficiently. This takes the form of a single parameter simulation run with multiple geometries and averaging on the result of each. Theory outlines that we must reach a point where an individual simulation result does not cause large deviation from the summed average of those before it; this is the point of convergence. On serial or low-powered computer systems however, this study is not required as the solver sequence can determine convergence after each iteration.

Once guidelines for the desired geometry of the model are known, static parameters are input to the program outlining specific properties of the surface being modelled. Here, given we recreated a surface covered with SiNWs, scanning electron microscopy (SEM) images were used as a basis for these parameters. These images are shown in figure 1.



(a) (b)

Figure 1: Scanning electron microscopy (SEM) images of a silicon surface decorated with vertically aligned nanowires from side-on (a) and top-down (b). This sample was fabricated and imaged for the work reported here.

Average values for the dimensions of the SiNWs in the model geometry are used as a point-of-reference to which randomisation can be applied. For instance, from the images shown in figure 1, SiNWs in the fabricated sample have an approximate height (h) of 1 µm, approximate average separation (pitch, g) of 180 nm and diameters (d) of up to 160 nm. Whlst each SiNW is different, these average values are representative of the properties across the entire surface of the sample. Some degree of pseudo-randomisation can then be applied to each SiNW within the model geometry. Each of these properties are varied on a per-SiNW basis using the following formulation:

$$z(S) = z_0 + \Delta z(S) \quad (1)$$

Here, z is the randomised parameter, z_0 is the average value for that parameter as outlined previously and Δz is some random variation to be applied dependant on a given input seed, S. Each parameter is varied within a set range. This is carried out by defining Δz as a random function with a mean of zero and a range of some definite amount. The model reported here applied the variation limits shown in table 1. Uniform functions were employed over Gaussian ones to enable greater control and predictability in the variation of a given static parameter.

Parameter, $z(S)$	**Average Value, z_0**	**Applied Variance, $\Delta z(S)$**
Radius (0.5d)	50 nm	$-30\text{ nm} \leq \Delta z \leq 30\text{ nm}$
Height (h)	1 µm	$-15\text{ nm} \leq \Delta z \leq 15\text{ nm}$
Pitch (g)	180 nm	$-56\text{ nm} \leq \Delta z \leq 56\text{ nm}$

Table 1: Variations applied to the key parameters of each silicon nanowire.

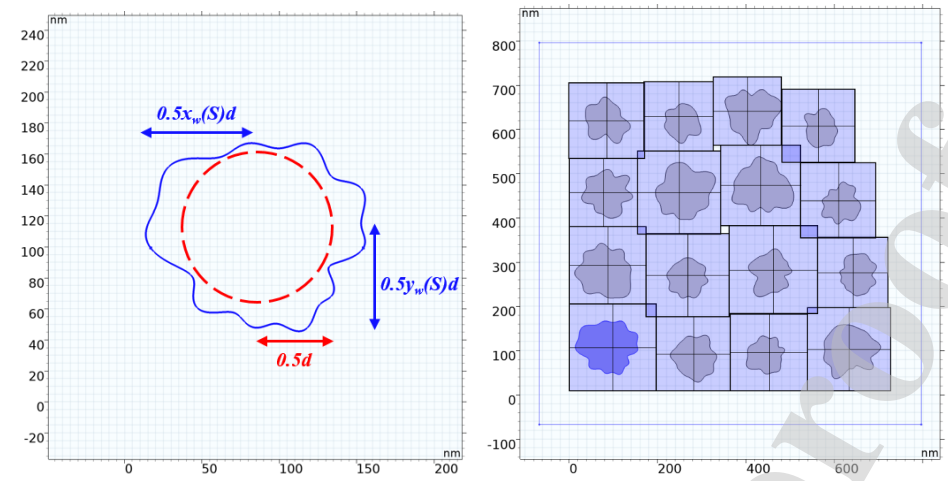
The values shown in table 1 are derived from SEM images of our manufactured SiNWs. Commonly, we have observed little deviation in height given a well-executed etching process. However, pitch and radius vary considerably more than this, and subsequently larger magnitudes of applied variance were permitted for these properties. Generating the SiNWs within the model geometry required a multi-step construction process consisting of several sub-elements. Notably, we needed to create a footprint, spine, and extrusion for each SiNW generated within their respective array. Experimental observations of real variants of these surfaces have demonstrated how such structures may benefit from not simply being modelled as cylinders extruding from the surface on which they are situated. For example, Scheul et al. [23] have demonstrated that many SiNWs have unpredictable outlines presenting an additional route for randomisation of a different type here.

Random footprints for these SiNWs were generated using a different random function coupled with a parametric curve element. This curve defaulted to unity at the scale set for the model; the nanometre range here. Subsequently, the parametric curve was initially generated about an approximate radius of 1 nm. This was scaled-up to the correct size as defined by the randomised radius variable $r_{0.5d}$ for that SiNW by multiplying the two together. This mathematical method is shown for both the x and y components of the curve in equations (2) and (3), where S is the input seed, r_u is a uniform random distribution, r_g is a Gaussian random distribution, N is the spatial frequency resolution and b is the spectral exponent.

$$x_w(S) = r_{0.5d}(S)\cos(2\pi s)\left[1 + 0.1\sum_{i=-N}^{N}\left[i^{-b}r_g(i,S)\cos\left(2\pi i s + r_u(i,S)\right)\right]\right] \quad (2)$$

$$y_w(S) = r_{0.5d}(S)\sin(2\pi s)\left[1 + 0.1\sum_{i=-N}^{N}\left[i^{-b}r_g(i,S)\cos\left(2\pi i s + r_u(i,S)\right)\right]\right] \quad (3)$$

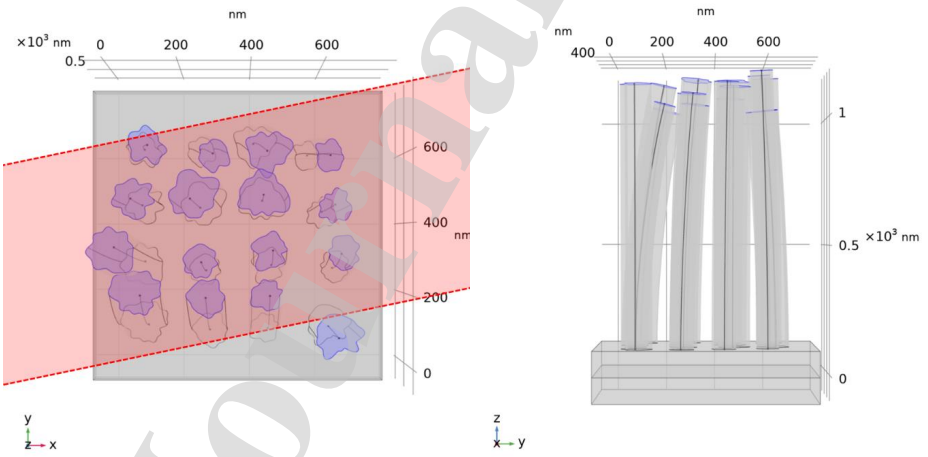
The values used in equations (2) and (3) to create the random footprints reported in figure 2 were N = 10 and b = 0.7. For the uniform random distribution, r_u, a mean of zero and a standard deviation of π was used, whilst for our Gaussian distribution, r_g, a mean of zero and a standard deviation of one was used. Figure 2a illustrates the relationship between the statically defined radius 0.5d and the randomised parametric curve about it after using the radius as a multiplier. In this form, it is important to realise that the curve is simply a boundary, and subsequently cannot be extruded into 3-dimensional nanowire domain directly. Instead, the convert-to-solid (CTS) geometry method within COMSOL Multiphysics®† was used to fill the area enclosed by this curve and subsequently visualise it as a surface instead. The use of this method required the implementation of a separate work plane, consisting of the newly-created surface, mapped to the original 3-dimensional domain. This meant that each SiNW footprint was not created within the original coordinate system but could be referenced from it. The work plane had a 2-dimensional Cartesian coordinate system denoted by x_w and y_w and held the footprints for all SiNWs within the array, inclusive of their variable pitch as shown in figure 2b.



(a) (b)

Figure 2: The domain-mapped 2-dimensional work plane hosting an exemplar single parametric curve-based boundary (a) and a 4x4 array of SiNW faces demonstrating variable pitch g and radius 0.5d (b).

The 3-dimensional extrusion of the surfaces shown in figure 2b is achieved externally from the work plane in the core simulation domain. These surfaces can be addressed individually and formed into nanowires using a Bézier polygon configured as a quadratic curve as a basis for extrusion. It is to this curve where the variation of height h must be applied, as well as any bend/bunching properties. In reference back to figure 1, it is clearly visible that these nanoscale structures tend towards one another at sufficient height. Being able to recreate this in simulation enables a further increased level of accuracy and enables us to observe how this effect changes other properties of the surface. An imaginary line is configured to be drawn between any two parallel edges of the model. Shown in figure 3a is this line bearing right whilst originating from the left. Using this line as a basis, a radius around it can be drawn to create an area of oblong proportions; this is the bunching zone. Any one of the 16 SiNWs created here, whose base sits within this zone, will be influenced towards this imaginary line laying directly in the centre of the zone each to varying degrees and with slightly different directions.



(a) (b)

Figure 3: Demonstrating how bending and bunching is achieved between the bottom and the top (highlighted in blue) of each SiNW. The tops can be seen to have the same shape as their

respective footprints but have been translated toward a bunching zone shown in red (a) following their Bézier polygon spines (b).

By generating enough different iterations of this geometry, we can simulate each whilst taking an average of all of them to more accurately obtain the property we are trying to simulate. In the case of this work, surface reflectance was taken as the property of interest.

3) Software Configuration and Optimisation

COMSOL Multiphysics®† is a finite element analysis tool, used in the completion of this work, employing various direct and/or iterative solvers to compute discretised partial differential equations (DPDEs). Partial differential equations (PDEs) are infinitely regressive and representative of real physical geometries existing in both space and time. Analytically near impossible to solve, discretisation needs to be introduced to these systems to ensure that there are only a finite number of elements making up an infinitely complex geometry. Conversion from a PDE to a DPDE makes the problem solvable in real-time. The solution accuracy for such DPDEs is highly dependent on the level of discretisation used within a given solver sequence. Studies were carried out to determine the minimum mesh element size required to accurately render our model, prevent deformation and to minimise the risk of unnecessary over-computation. On completion of this study, the solver generates a mesh and superimposes it over the un-discretised geometry. As demonstrated in figure 4, this will always introduce some aliasing into the model regardless of the mesh's density.

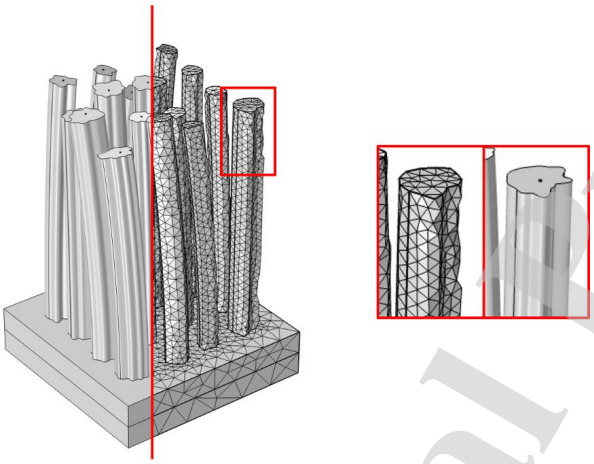


Figure 4: A direct comparison of a single unit-cell product of our randomisation and constraint pre-processing sequence, once it has been generated in the second step, before and after its mesh has been superimposed.

Once the entire geometry is meshed, it becomes possible to simulate it. Here, we employed the EM waves, frequency domain (EWFD) interface within Wave Optics Module of COMSOL®† to determine the reflectance of our geometries. A FEM model exhibiting such a high level of mathematical dependency on extra-geometrical components requires a considerable number of pre- and post-processing steps. As stated previously, the model was broken-down into key processing stages and executed in the appropriate sequence. It was designed to be as efficient on resources and accurate as possible. The randomisation of individual geometric features could be done anywhere in the pre-processing sequence. Despite this, we decided that it was crucial for all randomisation to occur before any rendering took place; significantly saving on computational resources and time. The implication here being that rendering a non-randomised geometry, only to then have it randomised and subsequently re-rendered, was undesirable, particularly on computer systems that lacked any graphics processing units (GPUs). Incidentally, this simulation is tailored to run remotely on the IRIDIS 5 supercomputer[1], notably lacking any form of GPU on its

[1] The IRIDIS Compute Cluster. https://www.southampton.ac.uk/isolutions/staff/iridis.page

standard compute nodes. A system-wide overview was drawn-up to ensure the potentially resource heavy simulation was designed to utilise the strengths of each system involved within the simulation sequence.

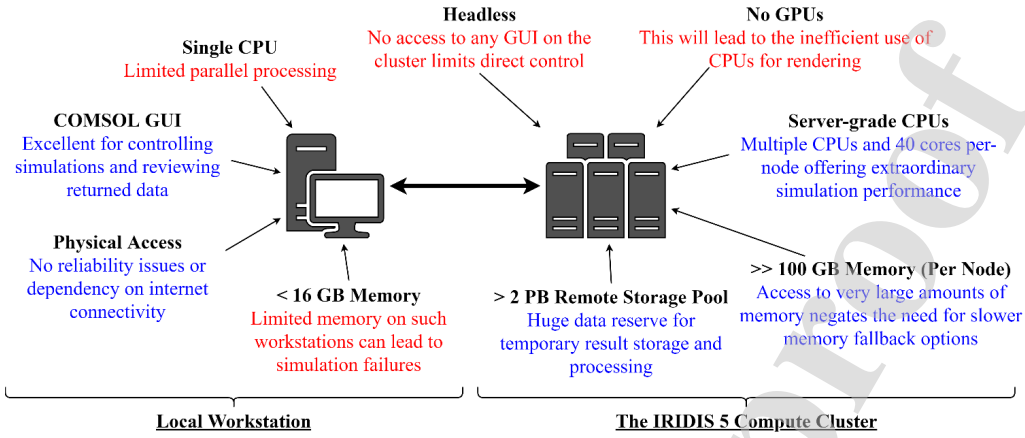


Figure 5: An overview of system specifics in preparation the optimised modelling workflow.

It is clear from the information shown in figure 5 that the local and remote systems each have their own strengths and weaknesses in general. By using each to showcase their advantages, and play-down their disadvantages, a simulation sequence suitable for both systems could be realised. Firstly, key to note were the only real disadvantages of using the cluster computer – no GPUs, and no user interface. All simulations with a 3-dimensional geometry, particularly a complex one, should both be rendered and meshed using a GPU. It is important to note that CPUs can be used for these tasks, but are not optimised to perform them, and subsequently can easily introduce unnecessary runtime and memory consumption, potentially crashing the solver mid-simulation. Given the local workstation had a GPU, it was decided to complete all rendering on this system, before passing it across to IRIDIS 5 for computation.

Simulations were configured to run using the parallel direct sparse solver (PARDISO), ensuring that out-of-core mode was switched off. This is because the nodes on IRIDIS 5 do not have any directly accessible local storage space, and subsequently out-of-core mode (OCM) would cause the simulation to crash on the event of a memory overload. Even if the nodes had local storage space, it is generally inadvisable to use the OCM. Writing data from memory to solid state or mechanical storage, and recalling it when needed, adds substantial delays to the solver sequence. It also can cause serious degradation to solid state storage devices, potentially leading to early component failure. This can simply be bypassed by preventing the solver from using OCM from the beginning, though it is essential to have a system with a significant amount of memory before disabling this mode. Each node on the cluster used here has 192 GB of high-speed Double Data Rate 4 (DDR4) memory – plenty for the simulation of even the most complex geometries explored here.

COMSOL®† offers a wide variety of options when it comes to solving remotely. Two software components needed to be configured for this sequence to run successfully; invoking commands and queue management. For COMSOL®† to invoke commands remotely, the "Remote Computing" preferences needed to be configured according to the cluster's setup. The local workstation ran a Windows®‡ 10 operating system not native to the secure shell (SSH) environment used by Linux systems such as the one on IRIDIS 5. A third-party program, "PuTTY", was used to interface with the cluster via SSH. "Multicore and Cluster Computing" preferences also needed to be set. IRIDIS 5 employs a SLURM®§ queuing system which, by default, COMSOL®† will not correctly interface with to provide the maximum amount of resources available on each node. Additional scheduler arguments needed to be added to ensure that all 40 cores on each node were available to the remote COMSOL®† kernel, and that the *'walltime'* was set appropriately. Simulation *'walltime'* is simply the amount of time the node will be available for use, and the simulation must subsequently finish within this time allocation. Furthermore, the queue name, *'batch'* as defined here, needed to be set to place the simulations in-line for the correct nodes within IRIDIS 5.

4) Simulation Methodology

A study first needed to be completed outlining how many different model geometries needed to be simulated in order to see convergence in our results. This took the form of a cluster sweep where the input seed S was varied from one to 50 in increments of one. A single, mid-band wavelength of 700 nm was injected into each geometry and the surface reflectance was measured. Each of these values for reflectance were then to be averaged together and the cumulative average plotted against the number of geometries forming that average. The point of convergence n_c could be seen when the cumulative average stabilised within a relative tolerance of 5%.

Taking this value forward, a full wavelength sweep would be performed on each of the geometries formed by the input seed S representative of values between one and n_c in increments of one. For instance, if the point of convergence n_c was found to be 15, then the reflectance response of 15 geometries would be averaged together with input seeds S of one to 15. On top of this, four levels of randomness were simulated so the effect of heterogeneity could be determined on the reflectance of a given surface. Examples of these levels can be seen in figure 6. It is important to note that, for purely periodic geometries, the point of convergence n_c is irrelevant and only one geometry needs to be simulated due to a lack of any randomisation. This is the only type of geometry that is completely independent of the input seed S.

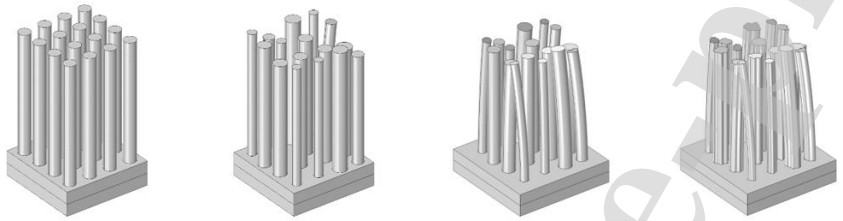


Figure 6: Demonstrating how the different *levels of randomness* are characterised as discrete levels and how they appear in the form of silicon nanowires.

For each of the levels shown in figure 6, simulation statistics such as runtime, memory usage, total element count and absolute error will be noted and compared to identify which is the best compromise between accuracy and resources available.

5) Results and Discussion

Convergence in the reflectance data is apparent after averaging around 39 different geometries (see figure 7). This is where the data holds a consistent relative tolerance of approximately ±2%; below the magnitude of 5% (±2.5%) stated in our simulation methodology. This represented an acceptable degree of uncertainty and was carried forward into the following geometric studies. This convergence point only appears to be affected by the geometric complexity of the model, whereby greater degrees of randomisation and variance between the simulated geometries results in more iterations required for convergence. This point is wavelength independent as we are using a fixed meshing strategy tailored for the smallest wavelength of light simulated; 400 nm in this work. Adaptive meshing would allow for the mesh to be regenerated based on the input wavelength, resulting in faster runtimes at larger wavelengths, however this was not the focus of this work. The convergence point is lower for less complex geometries, therefore the convergence data presented in figure 7 is for the most complex geometry, randomness level 3. Unfortunately, given the random nature of the geometries being simulated here it is not conceivable to predict the convergence point mathematically. Implementations of such a modelling approach must include a corresponding study to find this point, ensuring that further data extracted from simulations is accurate.

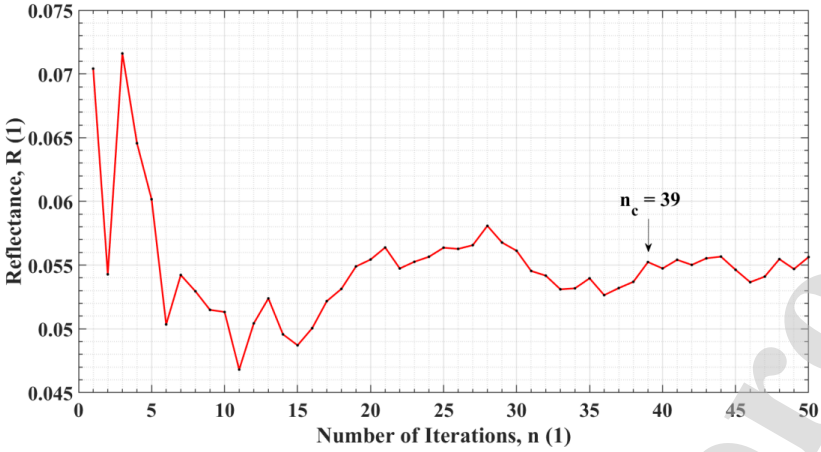
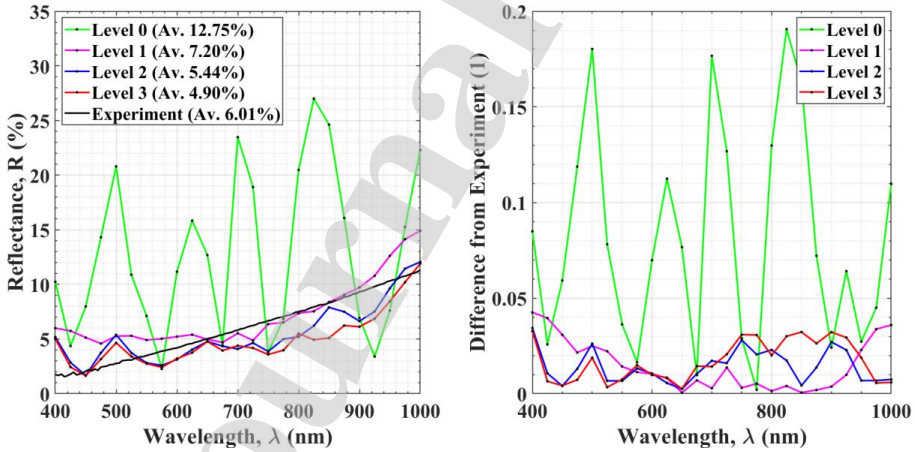


Figure 7: Convergence studies demonstrating stable reflectance measurements for a nanotextured surface, with randomness level 3 and 400 nm light input, at and beyond 39 random iterations.

In summary of the reflectance data gathered from our four simulations, and in comparison to a real sample, figure 8a showcases the immediately noticeable inaccuracy of taking a purely periodic approach to creating a SiNW geometry. Whilst it appeared the periodic SiNWs was formed primarily of noise, it was still possible to observe how it oscillates about the measured data. Such real SiNWs are highly anti-reflective due to their unique ability to create many different EM interference modes. In periodic structures, these interference patterns are not substantial enough to cancel one another out, giving a relatively flat-line response. Instead, only a few modes are apparent and their subsequent interference patterns are much more noticeable.

On the introduction of a slight degree of randomness, with each SiNW exhibiting varying pitch, radius, height and bend, there is a considerable decrease in this oscillatory effect. There is, much like in the bunched cylindrical nanowire geometry, a peak in reflectance observable at a wavelength of around 850 nm.



(a) (b)

Figure 8: Reflectance data for the four levels of randomness explored here and an equivalent real sample (a), shown alongside the same data's absolute error, excluding periodic nanowires, with respect to the real reflectance data (b). Broadband average (Av.) reflectance is also in the figure legend (a). Each simulation is the average of 39 different geometries.

A substantial finding, representative of the data shown in figure 8b, was that randomness level 1 appeared to be the most accurate of all the geometries studied here. In theory, this should not be the case. A greater level of randomness within the simulation geometry is indicative of a closer match between theory and structures we observe. There are several possible reasons behind why our results appear to show the opposite:

- Each randomised property is constrained within specific variance tolerances (see table 1);
- Generation of the model *'by eye'* using figure 1 introduces human error;
- Mesh inadequacy.
- Exclusion of the thin film aluminium oxide layer around the SiNWs.

It was likely that the error between simulation and experimental data, regardless of the randomness level, was derived from human error. As outlined in our modelling methods, we used an SEM image of real SiNWs to generate an accurate model of structures with similar heights amongst other properties. When transposing data from this imagery to our simulation domain, many of the properties were assumed to be approximate to certain values; each of which are outlined in table 1. These values were chosen based on SEM images of the fabricated structure. Whilst the height of the fabricated structures is relatively uniform, their pitch and radius tend vary to a much greater degree. This led to the implementation of the limits outlined for this work. The issue that arises from this is that properties such as radius and pitch are very difficult to quantify specifically, and the variation ranges are very much approximated based on only the small area of the sample imaged. Further variance and non-uniformities may be present elsewhere in the sample, within the area from which reflectance was measured. Also, given that these variation limits are defined as a proportion of the property they are varying, in opposition to a more statistical approach from real data, an element of systematic error is also introduced here.

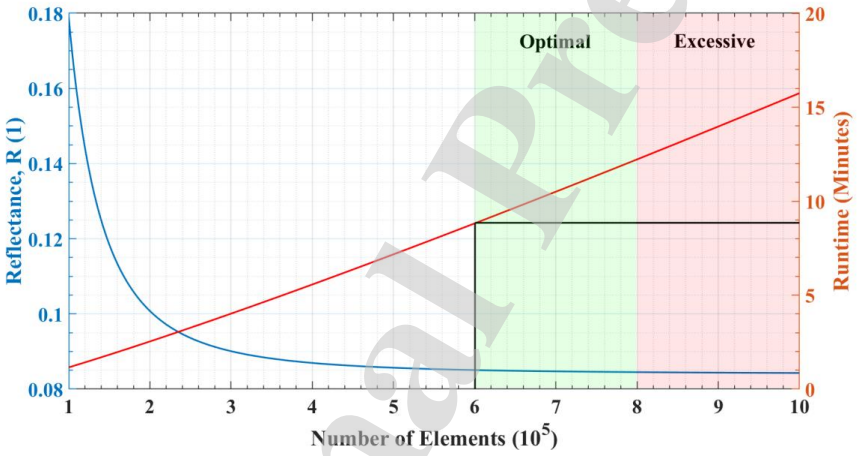
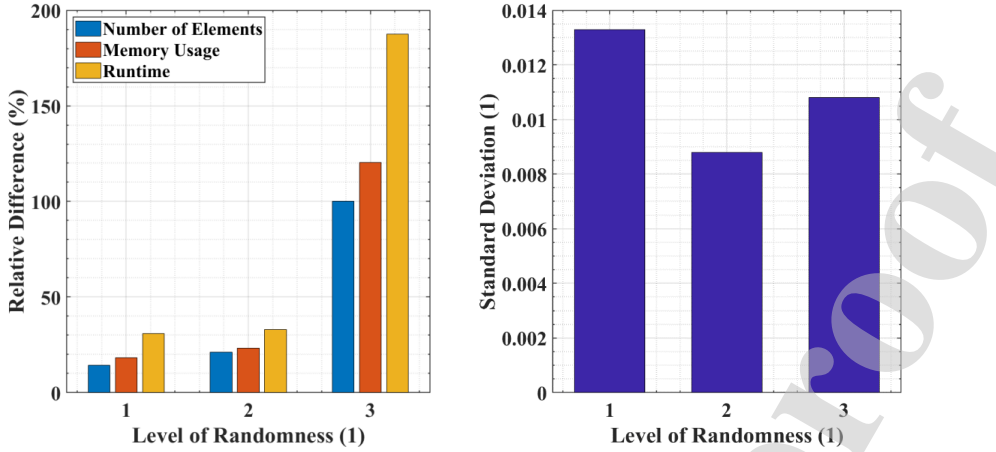


Figure 9: Data showcasing how the simulated reflectance for randomness level 3 varies given different mesh densities.

Mesh inadequacy, whereby the mesh density is insufficient to produce comprehendible data, is a common source of problems in modern simulation methods, particularly the FEM. Before commencing our reflectance studies we ensured our mesh density was enough to validate our data and rule this out as a potential cause for any differences observed between simulated and measured. Figure 9 demonstrates how, after a total element count of approximately 600,000, we see convergence on a reflectance value of 8.4%. As we further increase the element count, simulation accuracy changes very little but runtime continues to rise linearly. The implication here is that for any further increase in mesh density, we are only extending our runtime, yielding exactly the same data. It should be noted that for an element count of 800,000 and above, denoted here as 'excessive', the simulation becomes highly unstable and crashes frequently due to its memory demands. An 'optimal' mesh, used in all geometries yielding the results reported in figure 8, and consisting of 694,391 elements, is shown mapped over our periodic geometry in figure 9.

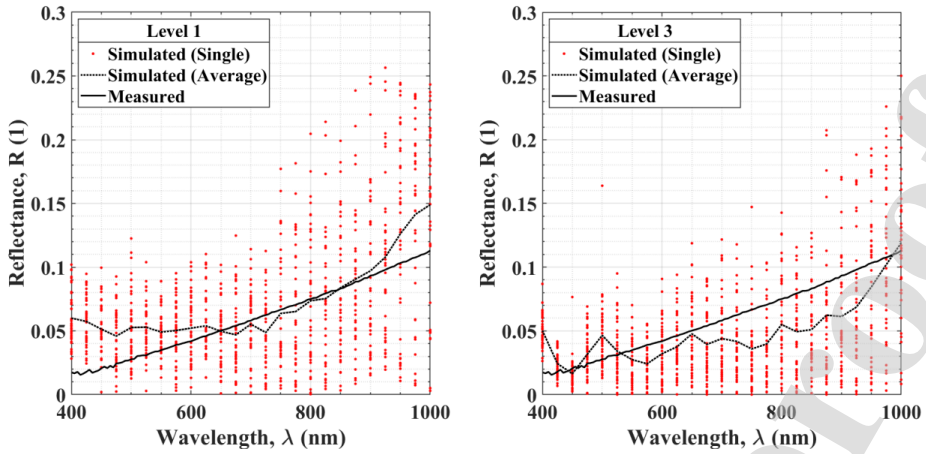


(a) (b)

Figure 10: Graphs showing the four key metrics for each randomness level as illustrated in figure 6. Level 0 had 279,579 elements, 35.31 GB of memory usage and a runtime of 41.57 minutes. Levels 1, 2 and 3 are reported relative to these values (a) as well as the standard deviation of the geometries' results across the 39 iterations (b). This data is indicative of a single wavelength of 400 nm, not a whole wavelength sweep.

It is noticeable that there is a strong correlation between the number of elements within a model (the mesh density), and the subsequent memory usage during the solution process. In fact, change in the differences between the levels as shown in figure 10a yields an essentially linear relationship. The implication here being that memory usage per-element appears to be independent of the element's size, and subsequently we can extrapolate an approximate memory requirement for much larger models under the same meshing strategy. The data here shows a memory requirement of approximately 150 kB/element. For the case of a slightly larger 5×5 SiNW periodic geometry with an identical mesh, approximately 1.56× larger than the equivalent 4×4, this would yield a memory requirement somewhere in the region of 58 GB. It is interesting to note that simulation runtime does not have quite as strong a correlation, and that of the level 3 geometry took considerably longer to simulate given the requirements in comparison to the others.

The trends shown in figure 10b have interesting implications for this investigation. The higher error of what was, in theory, the most accurate representation of a SiNW surface (randomness level 3), showed that the effort and higher resource requirements for this level of randomisation actually damages the accuracy of our model. Standard deviation metrics appear to showcase a greater level of convergence for level 2 against its average as shown in figure 8. This supports the argument that the properties of this particular geometry, notably the bunching tendencies, cause greater convergence on a specific reflectance value regardless of the state of the geometry. In effect, what can be inferred from this data is that bunching minimises the effect of heterogeneity on reflectance for small surface areas and causes greater uniformity in spectral response over a larger area. The distribution of individual simulation results can be seen for both of these levels in figure 11. The most accurate simulation model, however, was that of randomness level 1. Mid-range wavelengths from approximately 600 to 940 nm exhibited a match to experimental data with an average difference of only 0.05%, a first for the optical simulation of such structures. However, the shortfall of level 1 is apparent in the short and long wavelengths in the sweep, where both show a far higher reflectance than expected, overshooting by as much as 5% in these regions. In contrast, despite being slightly less accurate overall, levels 2 and 3 demonstrated a significant improvement in these regions, with errors of as little as 0.6% in the near-ultraviolet and 1% in the near-infrared. This implies a possible mixture of bunched and non-bunched SiNWs on our measured surface, and showcases the effect of bunching on surface reflectance outside the visual range.



(a) (b)

Figure 11: Demonstrating the distribution of single simulation results between level 1 (a) and level 3 (b).

Simulation efficiency is a measure of overall resource usage over time. A higher efficiency is indicative of a simulation that uses as much of the resources made available to it for as much of its total runtime as possible. This is often an important factor when requesting supercomputer time, and models with a greater efficiency are more desirable on such large scale, high power systems like that of IRIDIS 5 used here. It is worth noting that the efficiency of our model was relatively high, holding consistently at a value of approximately 73%.

6) Conclusions

Our model has helped showcase the dependencies of random variations within nanoscale structures on optical modelling. Furthermore, it has demonstrated an effective method of creating far more advanced levels of randomisation within what would otherwise be limited to static geometries. The most comparable work to that demonstrated here was conducted by Altermatt et al. in 2009 [21]. Their work reported a slightly different method of pseudo-randomisation used to achieve recreations of SiNWs that had not been explored prior. Due to the limited computational resources available, alongside a less diverse modelling environment, the level of geometric pseudo-randomisation as demonstrated in this work was not possible. The approach to recreating SiNW geometries explored here enables us to vary not only pitch, radius and tilt, but also bend, shape and bunching. This has opened new possibilities for identifying the influential factors of nanoscale surface properties on EM wave propagation.

Interestingly, it is apparent from the data presented here, that although introducing some heterogeneity results in significantly better agreement with experiment, extending the degree of randomisation does not always result in further improvement in accuracy. Overly complex models, such as randomisation level 3, turned out to be less accurate than its simpler counterparts. This shows that there is a balance , at least for SiNWs, between model complexity and reality, supporting the use of simplified models where intricate features may have little to no impact on the optical properties.

Aside from this specific geometry, the mathematical methods used to generate it are applicable to a wide range of different structures other than those explored here. For example, protein design and structure prediction, material or environmental feature growth and weather pattern simulation. Using the same mathematical methods, more variation can be added or even removed from an arbitrary geometry as required, making this a robust and versatile method of pseudo-randomisation. We have shown how the bunching of SiNWs affects reflectance, and also how inducing variations between surface features yields a more accurate response in this case. Despite also varying the shapes of the SiNWs, little improvement was seen at the cost of a 300% increase in simulation runtime, and 150% increase in resources, leading to the conclusion that randomisation, much like mesh density, has its limits.

There are disadvantages associated with this method however, which relate to the dependency on the FEM. This modelling methodology is notoriously resource and process intensive, making this level of randomisation only available for models created with access to sufficiently powerful hardware to run it within an acceptable amount of time. Furthermore, though the process workflow has been optimised for supercomputers here, some tweaking may be required in order to get higher efficiencies between different systems; the implication being that this is far from a simple plug-and-play program, and the mathematics must be carefully integrated into a study. Adaptive meshing methods can also be employed to reduce mesh density and simulation runtime as required for a variety of applications, include those explored here. Further study into this option is required however, as adaptive meshing may result in significantly deformed geometries if not conditioned correctly alongside these randomisation algorithms.

7) Acknowledgements

[†]COMSOL Multiphysics is a registered trademark of COMSOL AB.

[‡]Windows is a registered trademark of the Microsoft Corporation.

[§]SLURM is a registered trademark of SchedMD LLC.

This work has been supported by the Centre for Doctoral Training in New and Sustainable Photovoltaics (EPSRC grant number EP/L01551X/1) and completed as part of Black Silicon Photovoltaics (EPSRC grant number EP/R005303/1). The author of this work also acknowledges the use of the IRIDIS compute cluster, and associated support services at the University of Southampton, in the completion of this work.

8) References

[1] F.A. Fernandez, Proceedings of 1995 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (1995). https://doi.org/ 10.1109/sbmomo.1995.509668

[2] F.L. Teixeira, IEEE Transactions on Antennas and Propagation (2008) 2150-2166. https://doi.org/ 10.1109/tap.2008.926767

[3] B.M.A. Rahman, Proceedings of 8th Mediterranean Electrotechnical Conference on Industrial Applications in Power Systems, Computer Science and Telecommunications (1996). https://doi.org/ 10.1109/melcon.1996.551316

[4] J.J. Tyson, T. Rahman, S.A. Boden, Proceedings of the 2019 COMSOL Conference in Cambridge (2019).

[5] R.R. Singh, N. Malviya, V. Priye, IEEE Photonics Technology Letters (2016) 2889-2892. https://doi.org/10.1109/lpt.2016.2624501

[6] I. Khodadad, N. Dhindsa, S.S. Saini, IEEE Photonics Journal (2016) 1-10. https://doi.org/10.1109/jphot.2016.2548469

[7] Y.D. Almoallem, M.J. Moghimi, H. Jiang, 2017 International Conference on Optical MEMS and Nanophotonics (2017). https://doi.org/10.1109/omn.2017.8051469

[8] R.B. Sadeghian, 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (2013). https://doi.org/10.1109/embc.2013.6610832

[9] Z.X. Chen, H.Y. Yu, N. Singh, N.S. Shen, R.D. Sayanthan, G.Q. Lo, D.-L. Kwong, IEEE Electron Device Letters (2009) 754-756. https://doi.org/10.1109/led.2009.2021079

[10] S.K. Srivastava, D. Kumar, P.K. Singh, V. Kumar, 2009 34th IEEE Photovoltaic Specialists Conference (2009). https://doi.org/10.1109/pvsc.2009.5411524

[11] S.-C. Shiu, S.-B. Lin, C.-F. Lin, Conference on Lasers and Electro-Optics (2010). https://doi.org/10.1364/cleo.2010.jtub5

[12] H. Savin, P. Repo, G. von Gastrow, P. Ortega, E. Calle, M. Garin, R. Alcubilla, Nature Nanotechnology (2015) 624-628. https://doi.org/10.1038/nnano.2015.89

[13] T. Rahman, R.S. Bonilla, A. Nawabjan, P.R. Wilshaw, S.A. Boden, Solar Energy Materials and Solar Cells (2017) 444-453, https://doi.org/10.1016/j.solmat.2016.10.044

[14] A.G. Nassiopoulou, V. Gianneta, C. Katsogridakis, Nanoscale Research Letters (2011). https://doi.org/10.1186/1556-276x-6-597

[15] C. Li, K. Fobelets, C. Liu, C. Xue, B. Cheng, Q. Wang, Applied Physics Letters (2013) 183102. https://doi.org/10.1063/1.4826930

[16] A.M. Gouda, N.K. Allam, M.A. Swillam, 2016 Photonics North (2016). https://doi.org/10.1109/pn.2016.7537973

[17] A.M. Gouda, M. Elsayed, N.K. Allam, M.A. Swillam, 43rd IEEE Photovoltaic Specialists Conference (2016). https://doi.org/10.1109/pvsc.2016.7750185

[18] K.A. Gonchar, A.A. Zubairova, A. Schleusener, L.A. Osminkina, V. Sivakov, Nanoscale Research Letters (2016). https://doi.org/ 10.1186/s11671-016-1568-5

[19] T. Rahman, S.A. Boden, IEEE Journal of Photovoltaics (2017) 1556-1562. https://doi.org/ 10.1109/jphotov.2017.2748900

[20] D. Wu, X. Tang, K. Wang, X. Li, Scientific Reports (2017). https://doi.org/ 10.1038/srep46504

[21] P.P. Altermatt, Y. Yang, T. Langer, A. Schenk, R. Brendel, 34th IEEE Photovoltaic Specialists Conference (2009). https://doi.org/10.1109/pvsc.2009.5411125

[22] K.-Y. Zhou, S.-W. Jee, Z.-Y. Guo, Y. Xiao, S.A. Moiz, S.-T. Lui, J.-H. Lee, 37th IEEE Photovoltaic Specialists Conference (2011). https://doi.org/10.1109/pvsc.2011.6186570

[23] T.E. Scheul, E. Khorani, T. Rahman, S.A. Boden, AIP Conference Proceedings **2147**, 050009 (2019). https://doi.org/10.1063/1.5123858

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: