Predictions of the conductivity of CNT-polymer composites by an artificial neural network

M.A.S. Matos, S.T. Pinho and V.L. Tagarielli

Department of Aeronautics, Imperial College London, SW7 2AZ London, UK

Keywords: Carbon nanotubes, nanocomposite, electrical resistivity/conductivity, finite element analysis, neural networks

Submitted to Scripta Materialia, January 2019

Abstract

Industrial applications of conductive polymer composites with carbon nanotubes require precise tailoring of their electrical properties. While existing theoretical methods to predict the bulk conductivity require fitting to experiments and often employ power-laws valid only in the vicinity of the percolation threshold, the accuracy of numerical methods is accompanied with substantial computational efforts. In this paper we use recently developed physically-based finite element analyses to successfully train an artificial neural network to make predictions of the bulk conductivity of CNT-polymer composites at negligible computational cost.

Main

Conductive composites of carbon nanotubes (CNT) and polymers enable a wide range of applications [1, 2] which require precise tailoring of their electrical properties. Existing theoretical methods [3, 4] to predict the bulk conductivity of such composites require fitting to experiments and often employ power-law approximations which are accurate only close to the

---

1 Corresponding author. E-mail v.tagarielli@imperial.ac.uk
percolation threshold. Numerical models [5-9] can be more accurate but require substantial computational effort.

The electrical conductivity of a CNT-polymer composite depends on the conductivity of the nanotubes and of the polymeric matrix, as well as on the conductivity of the junctions between adjacent CNTs. Previous studies [10] suggest that the last effect dominates and is governed by fluctuation-induced tunnelling electron transport [11]. All these effects can be explicitly modelled by finite element (FE) analyses simulating the electrical response of representative volume elements (RVEs), as recently demonstrated by the authors [9]. We briefly summarise this previous work: the RVEs consisted of cubic volume elements where nanotubes of length $L_{\text{CNT}}$ and diameter $D_{\text{CNT}}$ were randomly dispersed until a target volume fraction $\nu_f$ was reached. Geometric periodicity was enforced and a search for intersecting CNTs was performed. Since the nanotubes are impenetrable, splines representing their centre-line were locally perturbed to enforce a minimum separation distance always larger than the van der Waals equilibrium distance (of 0.34 nm).

The electrical analysis was performed using Abaqus steady-state heat-transfer analysis [12] replacing all thermal properties with their electrical equivalents [13]. CNTs were discretized by DC1D2 2-node link elements embedded into a regular tessellation of DC3D8 8-node brick elements that represented the polymeric matrix. The conductivity between adjacent CNTs was modelled with a user element developed in a Fortran UEL subroutine of Abaqus [14]. This element reproduced the fluctuation-induced tunnelling electron transport [11] using Simmons’s generalized formula [15] and was applied to every contact point between distinct nanotubes, with distance inferior to 4 nm. The resistance across the nanotube junction depends on the separation distance $s$, on the insulating layer permittivity $\varepsilon = \varepsilon_r\varepsilon_0$ (where $\varepsilon_r$ and $\varepsilon_0$ are the relative and vacuum permittivity, respectively), and on the potential barrier $\varphi_0$, here taken as the CNT work function. Periodic boundary conditions were enforced at every boundary of the (periodic) RVE and 3 load steps were created, each representing a homogenized electric field – as illustrated in Fig. 1 – along each of the 3 Cartesian directions. The full conductivity matrix was computed by prescribing the homogenized electric potential difference and calculating the homogenized current that flows through the RVE.
This FE simulation technique is used here to produce training data for an artificial neural network (ANN). We conduct simulations of the conductivity of different CNT-polymer composite systems by adopting different values for the conductivities of CNTs and matrix, the length and diameter of the CNTs and the CNT volume fraction of the composites. The choice of the ranges of these values is inspired by studies appeared in the literature. The resulting percolation curves, representing the homogenized isotropic electrical conductivity $\bar{\kappa}_c$ as function of the volume fraction are presented in Fig. 2 for illustration. We note that each datapoint in Fig. 2 is obtained via Monte Carlo analyses and it represents the average of 20 repeated simulations conducted on 20 realisations of the microstructure.

A dimensional analysis of the physical problem at hand is now performed, applying the Buckingham Pi theorem [16]. The objective is to reduce the number of problem parameters, to facilitate training of an ANN. For a given volume fraction $\nu_f$, the macroscopic (homogenised) bulk conductivity $\bar{\kappa}_c$ will be a function of the nanotube dimensions $L_{\text{CNT}}$ and $D_{\text{CNT}}$, matrix $\kappa_m$ and nanotube $\kappa_f$ conductivities, matrix permittivity $\epsilon = \epsilon_r \epsilon_0$ and nanotube work fraction $\varphi_0$, assuming that the FE predictions are mesh-independent (which was ensured by choosing an adequate mesh after a preliminary mesh convergence study). The above variables contain 4 distinct fundamental physical dimensions: mass $M$, length $L$, time $T$ and current $I$, which define the set $\mathbb{D} = \{M, L, T, I\}$. To allow the definition of a set of dimensionless quantities that include all the independent variables, Planck’s constant $h$ (featuring in Simmons’s formula [15]) is added to the list of dimensional parameters, resulting in a set of 8 dimensional quantities: $\mathbb{X} = \{\bar{\kappa}_c, \kappa_f, \kappa_m, L_{\text{CNT}}, D_{\text{CNT}}, \varphi_0, \epsilon, h\}$. 

Fig. 1. Colour map of the electric potential field in an RVE.
Fig. 2. Homogenized conductivity curves for different CNT-polymer composites ($\epsilon_r = 3.98, \varphi_0 = 4.95$ eV).

The units $[x_i]$ of each dimensional variable $x_i \in \mathbb{X}$ can be written as a product of the base dimensions $d_j \in \mathbb{D}$, as follows

$$[x_i] = \prod_{j=1}^{4} d_j^{\alpha_{ij}}$$

where $\alpha_{ij}$ represents the exponent of the fundamental dimensions $d_j$ to obtain the units of variable $x_i$.

Defining a new non-dimensional variable $\Pi_k$ as a product of the initial dimensional variables gives...
\[ \Pi_k = \prod_{i=1}^{8} x_i^{\beta_{ik}} \]  

which has dimensions

\[ [\Pi_k] = \prod_{j=1}^{4} d_j^{\alpha_j \beta_{jk}} = \prod_{j=1}^{4} d_j^{\gamma_{jk}} \]  

Since \( \Pi_k \) is dimensionless and the base dimensions are independent, the exponent \( \gamma_{jk} \) must be zero. In matrix form, this means that

\[ \alpha^T \beta = \gamma \]  

needs to be null, and therefore the exponent matrix \( \beta \) is determined by finding the null space of \( \alpha^T \). For the current case, the dimensional matrix \( \alpha^T \) and respective null space \( \mathcal{A} \) are:

\[
\alpha^T = \begin{bmatrix}
-1 & -1 & -1 & 0 & 0 & 1 & -1 & 1 \\
-3 & -3 & -3 & 1 & 1 & 2 & -3 & 2 \\
3 & 3 & 3 & 0 & 0 & -2 & 4 & -1 \\
2 & 2 & 2 & 0 & 0 & 2 & 0 & 0 \\
\end{bmatrix}
\]

\[ \mathcal{A} = \begin{bmatrix}
\begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \\
\begin{bmatrix} 1 \\ 0 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \\
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ -1 \\ 0 \\ 1 \end{bmatrix}, \\
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 1 \end{bmatrix} \\
\end{bmatrix} \]  

It results that the physical problem investigated depends on only 5 non-dimensional groups, namely

\[
\Pi_0 = \frac{\bar{\kappa}_c - \kappa_m}{\kappa_f - \kappa_m}, \quad \Pi_1 = \frac{\kappa_f}{\kappa_m}, \quad \Pi_2 = \frac{L_{\text{CNT}}}{D_{\text{CNT}}}, \quad \Pi_3 = \frac{\kappa_f h}{\epsilon \varphi_0}, \quad \Pi_4 = v_f. \]  

We note that the algebraic form of the non-dimensional group \( \Pi_0 \) was chosen to limit the variability of its numerical value, which facilitates training of the ANN. The curves presented in Fig. 2 are now shown in non-dimensional form in Fig. 3. We note that the abscissa is selected...
as the product of the two non-dimensional parameters $\Pi_2 \times \Pi_4 = v_f L_{\text{CNT}}/D_{\text{CNT}}$ as it allows for a further simplified representation of these curves.

![Graph](image)

Fig. 3. Non-dimensional conductivity curves for different CNT-polymer composites, as a function of the volume fraction multiplied by the aspect ratio of the CNT.

The data in Fig. 3 is now used to train an artificial neural network (ANN) [17]. An ANN can be described as a collection of layers composed by nodes. The (real) value of each node is a result of a non-linear function $f_A$ applied to a linear combination of its inputs. The value of each node $i$ of layer $j$, $v_i^{(j)}$, is obtained from the values of the nodes in the previous layer, as

$$v_i^{(j)} = f_A\left(w_{ik}^{(j)} v_k^{(j-1)} + b_i^{(j)}\right)$$

where $w^{(j)}$ and $b^{(j)}$ are the layer’s weight matrix and bias array. These represent the system parameters which will be determined during the training procedure. The first layer of the ANN represents the inputs and the last layer represents the outputs. The sigmoid function is chosen as the activation of the inner layers, $f_A(x) = (1 + e^{-x})^{-1}$, and identity (no activation) is used for the output layer.

The network is assembled and trained using TensorFlow [18]. Due to their high variability, all inputs and outputs are replaced by their logarithm, which is then linearly scaled to the range...
according to its minimum and maximum numerical values (by the function \( \text{minmax}(\ ) \)). These operations allow for better distributing the training data within the considered range, facilitating the optimization procedure. The input \( \mathbf{X} \) and output \( \mathbf{Y} \) vectors are

\[
\mathbf{X} = \left\{ \begin{array}{c}
\text{minmax}\left( \log_{10} \frac{v_l L_{\text{CNT}}}{D_{\text{CNT}}} \right) \\
\text{minmax}\left( \log_{10} \frac{\kappa_f}{\kappa_m} \right) \\
\text{minmax}\left( \log_{10} \frac{L_{\text{CNT}}}{D_{\text{CNT}}} \right) \\
\text{minmax}\left( \log_{10} \frac{\kappa_f h}{\varepsilon \varphi_0} \right)
\end{array} \right\}, \quad \mathbf{Y} = \left\{ \text{minmax}\left( \log_{10} \frac{\kappa_c - \kappa_m}{\kappa_f - \kappa_m} \right) \right\}
\]

The network is trained by minimizing the loss function, defined as the mean square error of the predictions, by optimizing the parameters on which it depends (\( \mathbf{w} \) and \( \mathbf{b} \)) with backpropagation and the Adam algorithm [19]. To avoid overfitting, L2 regularization is used. This penalizes the loss function by the squared value of the weights scaled by \( \lambda_{L2} \), discouraging large weight values.

Results from 8500 simulations are used as training data, and an additional set of properties is used for validation. Several configurations were tried, and the best encountered – considering training and test accuracies – had the configuration [4 – 8 – 4 – 1], with two hidden layers of 8 and 4 neurons, respectively, using a small regularization with \( \lambda_{L2} = 2.5 \times 10^{-4} \). This resulted in an output mean square error of 1.23% and 0.34% for the training and validation data considered, respectively. The equation that represents the optimized ANN is presented in the Appendix, allowing the reader to make direct use of the findings of the current study.

To investigate the quality of the predictions of this ANN outside the training set, we considered the properties of two CNT-polymer composites available in the literature, with properties summarized in Table I. These represent multiwall CNTs in a Bisphenol-A epoxy matrix [20], or in an epoxy matrix [21]. Predictions using the trained ANN are compared in Fig. 4 to predictions of other authors. Clearly, good agreement is found with the predictions by Gong et al [20] (who use an equivalent resistor network) and by Feng and Jiang [21] (who use a micromechanical modelling approach).
The predictions of the ANN require a CPU time of about 7 orders of magnitude smaller than the detailed FE simulations of [9]. This suggests that ANNs are an effective tool to package complex information on material response.

The optimised ANN presented in equation (9) in the appendix can be readily used to make accurate predictions of the conductivity of any CNT-polymer composite system. In principle, the same ANN could also be employed to predict the conductivity of other types of composites (e.g. short-fibre composites) or similar conductive networks. Investigating this is left as a topic for future studies.

![Graph of Composite Conductivity vs. Volume Fraction]

**Fig. 4.** Predictions of the trained ANN using the same properties as those by a) Gong et al. [20] and b) Fend and Jiang [21].

**Table I.** Properties used for the validation examples. Values marked with * were not explicitly reported by the authors.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Diameter $D_{CNT}$</td>
<td>15 nm</td>
<td>20 nm</td>
</tr>
<tr>
<td>Length $L_{CNT}$</td>
<td>1.5 μm</td>
<td>20 μm</td>
</tr>
<tr>
<td>CNT conductivity $\kappa_f$</td>
<td>$10^3$ S/m</td>
<td>$10^4$ S/m</td>
</tr>
<tr>
<td>CNT work fraction $\varphi_0$</td>
<td>4.7 eV</td>
<td>5.0 eV</td>
</tr>
<tr>
<td>Matrix conductivity $\kappa_m$</td>
<td>$10^{-12}$ S/m*</td>
<td>$10^{-13}$ S/m</td>
</tr>
<tr>
<td>Matrix relative permittivity $\varepsilon_r$</td>
<td>3.0*</td>
<td>3.98*</td>
</tr>
</tbody>
</table>
Acknowledgments

This project was funded by the European Union’s Horizon 2020 research and innovation program, under the Marie Skłodowska-Curie [Grant Agreement no. 642890] ([http://thelink-project.eu](http://thelink-project.eu)).

References

The optimized neural network that relates the non-dimensional output to the non-dimensional inputs can be written as

\[
Y = b^{(3)} + w^{(3)} \times f_\Lambda \left( b^{(2)} + w^{(2)} \times f_\Lambda \left( b^{(1)} + w^{(1)} X \right) \right)
\]  \hspace{1cm} (9)

\[
w^{(1)} = \begin{bmatrix}
3.933 \times 10^{-1} & 1.410 \times 10^{-1} & -7.309 \times 10^{-1} & 1.153 \times 10^{-1} \\
5.779 & 3.361 \times 10^{-1} & 2.860 \times 10^{-1} & 2.436 \times 10^{-1} \\
2.280 & 4.842 \times 10^{-1} & -1.793 & -1.255 \times 10^{-1} \\
-3.024 & 3.451 \times 10^{-1} & -1.407 & 1.592 \times 10^{-2} \\
1.341 \times 10^{-6} & 4.618 \times 10^{-7} & -2.318 \times 10^{-7} & 2.253 \times 10^{-6} \\
5.493 \times 10^{-1} & -1.023 & 5.758 \times 10^{-1} & -1.845 \times 10^{-1} \\
2.324 \times 10^{-2} & -3.114 \times 10^{-1} & 6.348 \times 10^{-1} & -7.045 \times 10^{-2} \\
-9.107 \times 10^{-2} & -1.241 \times 10^{-2} & -6.064 \times 10^{-1} & 2.187 \times 10^{-1}
\end{bmatrix}
\]  \hspace{1cm} (10)

\[
b^{(1)} = \begin{bmatrix}
9.421 \times 10^{-1} \\
-6.326 \times 10^{-1} \\
2.071 \\
2.247 \\
-1.147 \times 10^{-1} \\
-1.500 \\
-4.005 \times 10^{-2} \\
-4.518 \times 10^{-1}
\end{bmatrix}
\]

\[
w^{(2)} = \begin{bmatrix}
-5.806 \times 10^{-1} & 9.842 \times 10^{-2} & 2.875 \times 10^{-1} & 4.949 \times 10^{-1} \\
2.500 & -1.452 & -6.793 \times 10^{-1} & 4.294 \\
-1.208 & 4.508 \times 10^{-1} & 6.955 \times 10^{-1} & 2.237 \\
-9.053 \times 10^{-1} & -6.235 \times 10^{-1} & -3.552 \times 10^{-1} & -2.928 \\
4.813 \times 10^{-4} & -1.715 \times 10^{-4} & -6.040 \times 10^{-4} & 8.194 \times 10^{-7} \\
8.292 \times 10^{-1} & 8.498 \times 10^{-1} & 4.240 \times 10^{-1} & 3.001 \times 10^{-1} \\
7.741 \times 10^{-1} & 7.446 \times 10^{-2} & 3.98 \times 10^{-3} & -2.672 \times 10^{-1} \\
-5.135 \times 10^{-1} & 1.964 \times 10^{-1} & 2.116 \times 10^{-1} & -2.966 \times 10^{-2}
\end{bmatrix}^T
\]  \hspace{1cm} (11)

\[
b^{(2)} = \begin{bmatrix}
1.219 \\
8.328 \times 10^{-1} \\
-2.781 \times 10^{-1} \\
-1.948
\end{bmatrix}
\]

\[
w^{(3)} = [-2.352 \quad 1.940 \quad 1.270 \quad 2.763]
\]

\[
b^{(3)} = [-1.269].
\]