Supporting Information

Quantitative 2D fitting of fluorescence-excitation maps: Excitation lineshape of single-wall carbon nanotubes

Sofie Cambré,[a,b‡] Wouter Van Werveke,[a‡] Miguel De Clercq,[a‡] Maksiem Erkens,[a] Miles Martinati,[a] and Wim Wenseleers*[a]

[a] *Nanostructured and Organic Optical and Electronic Materials, Physics Department, University of Antwerp, Belgium;* [b] *Current address: Theory and Spectroscopy of Molecules and Materials, Physics Department, University of Antwerp, Belgium;* [‡] *These authors contributed equally to this work;* [*] *E-mail: Wim.Wenseleers@uantwerp.be*

Contents

1 Experimental details for acquiring the PLE maps

All the two-dimensional (2D) IR PLE spectra shown in this work were recorded using a home-built setup. Each sample was excited with a pulsed Xe-lamp (custom adapted from Edinburgh Instruments, Xe900-xP920) and excitation wavelengths were spectrally selected with a 300 mm focal length grating monochromator (Acton SpectraPro 2355). Emission was collected at 90° and analyzed using a 150 mm focal length grating spectrograph (Acton SpectraPro 2156) with a liquid nitrogen cooled extended InGaAs photodiode array detector (Princeton Instruments OMA V:1024/LN-2.2) sensitive up to 2.2 µm. Spectra were recorded with 5 nm steps in excitation wavelength. Appropriate filters were used to eliminate stray light and higher order diffractions from the spectrometers and all spectra were corrected for detector and spectrograph efficiency, filter transmission, reabsorption within the cell and (temporal and spectral) variations of the excitation light intensity.

2 Fitting details

As mentioned in the main text of this publication, a Graphical User Interface (GUI) has been developed under the name PLEfit2D, which can be found at the PLEfit2D website https://plefit2d.uantwerpen.be. In this section, we explain different aspects of the implementation of our fitting model within this GUI, such as the exact procedure and algorithms used to perform the actual fit and to determine error bars on the model parameters.

2.1 Extracting the SWCNT diameter from a PLE map

As explained in the main text, we use slightly adapted versions of the empirical relations first derived by Bachilo *et al.*[1] to extract the SWCNT diameter from a PLE map, to then be used in the diameter-dependence of all phonon side band amplitudes as well as the position of the RBM phonon side band. These empirical relations are of the form:

$$\tilde{\nu}_{jj} = \frac{10^7 \mathrm{cm}^{-1}}{B_1^j + B_2^j d} + \frac{A_\mu^j \cos(3\alpha)}{d^2} \mathrm{cm}^{-1} \tag{S1}$$

with $\tilde{\nu}_{jj}$ the transition frequencies (in cm^{-1}) for the first ($j = 1$) and second ($j = 2$) optical transitions respectively and d the SWCNT diameter in nm. This equation contains 8 distinct parameters: two pairs (B_1^j, B_2^j) for each optical transition j and four A_μ^j parameters which are different for both optical transitions as well and also depend on the SWCNT modulus $\mu := (n - m) \bmod 3 = 1, 2$.

By treating d and α as continuous variables, (S1) can be inverted and the diameter d can be extracted from $\tilde{\nu}_{11}$ and $\tilde{\nu}_{22}$. The parameters in (S1) depend also on the specifics of the sample, such as the inner and outer environment of the SWCNTs.[2,3] As explained in the main text, this does not interfere with the accuracy of a fit with our model and we use a single set of parameters for the sake of consistency. In particular, we employ those obtained for empty SWCNTs in an aqueous solution with sodium deoxycholate. These parameters were first determined in reference 2 by fitting equation (S1) to peak positions obtained from 1D fits of different slices through the PLE maps. Later, the parameters were

refined by directly fitting the empirical relations to the PLE maps using this fitting model, including PLE maps of HiPco and ARC-discharge SWCNTs to provide a sufficiently broad diameter range in the PLE map.[3] Table S1 lists the resulting parameters.

Transition	B_1	B_2	$A_{\mu=1}$	$A_{\mu=2}$
$\tilde{\nu}_{11}$	147.67	1097.40	−636.12	318.35
$\tilde{\nu}_{22}$	145.60	581.37	1114.73	−1397.35

Table S1: Parameter values for equation (S1) as used in our GUI, determined for empty SWCNTs in aqueous solution with sodium deoxycholate.[3]

2.2 Background fitting

Remember that for a PLE map containing N distinct chiralities, our model consists of N two-dimensional (2D) basis functions F_i which are combined in a linear regression scheme to obtain amplitudes for each:

$$\mathbb{F}(E_{\mathrm{em}}, E_{\mathrm{ex}}) = \sum_i A_i F_i(E_{\mathrm{em}}, E_{\mathrm{ex}}). \tag{S2}$$

Additionally, to take into account any background in the PLE maps, a few (N_b) more basis functions can be added. Three different options for backgrounds exist in the GUI:

(1) no background: $N_b = 0$.

(2) a constant background which is implemented as a basis function $F_{N+1} = 1$, such that its amplitude directly yields the constant. In this case $N_b = 1$.

(3) a bilinear background in emission and excitation energy, implemented by including $N_b = 4$ additional basis functions

$$F_{N+1} = 1, \qquad F_{N+2} = E_{\mathrm{em}}, \qquad F_{N+3} = E_{\mathrm{ex}}, \qquad F_{N+4} = E_{\mathrm{ex}} \cdot E_{\mathrm{em}}.$$

None of these background basis functions add any additional parameters to the fitting model that need to be iteratively optimised, since the amplitudes of the basis functions are analytically determined by the linear regression scheme in equation S2.

2.3 Procedure & algorithms

The actual fit itself is performed by the MATLAB® function `lsqcurvefit` using a trust-region algorithm with upper and lower bounds on the model fit parameters. Each iteration generally consists of two parts: first the model parameters (such as peak positions and linewidths) are used to create a set of 2D basis functions, then their weights (amplitudes) are calculated in a linear regression scheme. In the first part upper and lower bounds on the parameters can be imposed, though good initial values are always key. The second part ensures that the optimal amplitudes are instantly calculated after each adjustment to the model parameters. However, care must be taken in this part because only non-negative values are physically relevant but standard linear regression does not impose this restriction. Although numerical convergence algorithms exist to solve the non-negative least-squares (NNLS) problem,[4] they are significantly slower and

lack the analytical robustness of standard linear regression. Since speed may be a critical consideration for some use-cases of our code, alternative ways of dealing with negative amplitudes are implemented in the algorithm and can be selected. These are discussed further down in this section.

The fitting is typically performed in batches of iterations, halting the fit after a selected maximum number of steps have been taken to allow for user inspection and, if necessary, intervention. Examples of this include peaks which have grown unphysically wide or migrated to coincide with another peak. This can be resolved either by manually adjusting the linewidth and/or position of a troublesome peak and letting the fit continue from there, or by removing the peak altogether if the problem persists. Once the fit stabilises after a number of those batches, it is allowed to converge without interruption. An important factor to consider throughout this procedure, is how to deal with the aforementioned possibility of negative amplitudes within a standard linear regression scheme. Our GUI provides two basic approaches, laid out below.

Adapted linear regression. This is the default algorithm for determining the optimal amplitudes and it consists of three steps.

1) A standard linear regression is performed, resulting in a set of amplitudes $\{A_i\}$.
2) If any of the amplitudes are negative, say those for $i \in \{\ell \,|\, A_\ell < 0\}$, we remove the basis functions F_ℓ from $\mathbb{F}$ and perform another standard linear regression including only the positive-amplitude peaks.
3) Repeat step 2) until no negative amplitudes result.

The output of this procedure is $\mathbb{F}$ including all basis functions $F_{i\neq\ell}$ with their amplitudes $A_{i\neq\ell}$ as calculated in the last iteration of step 2) and all basis functions F_ℓ with their amplitudes A_ℓ set to 0. This ensures that the resulting fit is not only free of unphysical negative amplitudes, but also a proper solution of the linear regression equations, getting rid of any incorrect compensation or cancellation of peaks and pushing the other parameters towards a physically accurate fit. Additionally, this approach is much faster than using the NNLS algorithm and moreover it remains analytical and hence robust.

Standard linear regression. Sometimes it might be useful or informative to retain the possibility of negative amplitudes in the full fit. For this reason, an option is given in PLEfit2D to determine the A_i using standard linear regression, without any attempt to remove or exclude negative amplitudes. This option should be used with caution.

2.4 Error calculation

For the purposes of this discussion, we consider $\mathbb{F}(E_{\mathrm{em}}, E_{\mathrm{ex}})$ as a long vector of length $n_{\mathrm{em}}n_{\mathrm{ex}}$, where n_{em} and n_{ex} are the number of data points in emission and excitation, respectively. We use linear indexing to denote the components of this vector as $\mathbb{F}_k$, *i.e.* each k corresponds to a unique pair $(E_{\mathrm{em}}, E_{\mathrm{ex}})$. We further denote the full collection of fit parameters related to lineshapes with $\mathbb{P}$, containing both the parameters shared among all basis functions F_i, such as the exponent in the diameter dependence of the RBM phonon sideband, and all parameters unique to a given peak F_i such as its position. This collection can also be considered as a 1D vector, the elements of which are denoted as $\mathbb{P}_\ell$. Each ℓ corresponds to a specific parameter.

2.4.1 Model parameters

The errors on the lineshape parameters $\mathbb{P}$, such as exciton peak width and position, are determined as follows:

(1) The residual variance σ^2 is estimated from the residuals $\mathbf{r}(\mathbb{P}) := \mathbb{F}(\mathbb{P}) - \mathbb{S}$ as

$$\hat{\sigma}^2(\mathbb{P}) = \frac{\|\mathbf{r}(\mathbb{P})\|^2}{n_{\mathrm{rdof}}}. \tag{S3}$$

Here $\mathbb{S}$ is the experimental signal vectorised in the same way as the fit $\mathbb{F}$, n_{rdof} is the number of residual degrees of freedom, discussed below, and $\|\cdot\|$ is the standard Euclidean norm.

(2) The covariance matrix $\Sigma(\mathbb{P})$ is then estimated as

$$\hat{\Sigma}(\mathbb{P}) = \left(J^T J\right)^{-1} \hat{\sigma}^2(\mathbb{P}), \tag{S4}$$

where

$$J_{k\ell} = \frac{\mathrm{d}\mathbb{F}_k}{\mathrm{d}\mathbb{P}_\ell} \tag{S5}$$

is the Jacobian matrix of $\mathbb{F}$ w.r.t. the lineshape parameters, as calculated by the MATLAB® function `lsqcurvefit`.

(3) The 1σ error margin on parameter $\mathbb{P}_\ell$ is then calculated as $\sqrt{\hat{\Sigma}_{\ell\ell}(\mathbb{P})}$.

Theoretically n_{rdof} is given by

$$n_{\mathrm{rdof}} = n_{\mathrm{dof}} - n_p,$$

with n_{dof} the overall number of degrees of freedom (DOF), given by $n_{\mathrm{em}} n_{\mathrm{ex}}$, and n_p is the number of fit parameters, given by the length of $\mathbb{P}$. However, the errors obtained in this way do not take into account systematic deviations of the data from the model for the fitted values of the model parameters, due to possible model inaccuracies.

It is possible to estimate the influence of systematic errors by examining the zero-crossings of the residuals. Indeed, without systematic error, the residuals are expected to be normally distributed around 0, which implies there are

$$n_0 = \frac{n_{\mathrm{em}} n_{\mathrm{ex}}}{2} = \frac{n_{\mathrm{dof}}}{2} \tag{S6}$$

sign changes or zero-crossings in the residuals. By counting the number of these crossings it is hence possible to estimate an effective number of degrees of freedom $n_{\mathrm{dof,eff}}$. If there is a systematic error, this number will be smaller than the theoretical n_{dof}: the residuals systematically deviate above or below zero. The effective number of DOF is thus estimated by twice the amount of sign changes in the residual, both in emission (n_0^{em}) and excitation (n_0^{ex}). Since there are n_{ex} emission profiles in a 2D PLE map and n_{em} excitation profiles, the effective number of degrees of freedom provided by each profile is given by

$$n_{\mathrm{dof,eff}}^{\mathrm{em}} = 2\frac{n_0^{\mathrm{em}}}{n_{\mathrm{ex}}} \tag{S7}$$

$$n_{\mathrm{dof,eff}}^{\mathrm{ex}} = 2\frac{n_0^{\mathrm{ex}}}{n_{\mathrm{em}}}. \tag{S8}$$

The effective number of total DOF is then given by

$$n_{\mathrm{dof,eff}} = n^{\mathrm{em}}_{\mathrm{dof,eff}} n^{\mathrm{ex}}_{\mathrm{dof,eff}} \tag{S9}$$

$$n_{\mathrm{dof,eff}} = \frac{4 n_0^{\mathrm{ex}} n_0^{\mathrm{em}}}{n_{\mathrm{ex}} n_{\mathrm{em}}}, \tag{S10}$$

and the effective number of residual degrees of freedom becomes

$$n_{\mathrm{rdof,eff}} = n_{\mathrm{dof,eff}} - n_p. \tag{S11}$$

2.4.2 Amplitudes

The same general approach is taken in calculating the error bars on the amplitudes. However, the Jacobian is now calculated explicitly as

$$J_{ki} = \frac{\mathrm{d}\mathbb{F}_k}{\mathrm{d}A_i} = F_{ki} \tag{S12}$$

where $F_{ki} = F_i(E_{\mathrm{em}}, E_{\mathrm{ex}})$ is the value of the i-th basis function at the point $(E_{\mathrm{em}}, E_{\mathrm{ex}})$ uniquely defined by the linear index k. Again the effective number of residual degrees of freedom $n_{\mathrm{rdof}} = n_{\mathrm{dof}} - N$ is estimated in the same way as for the model parameters. From this Jacobian the covariance matrix and ultimately the 1σ confidence intervals are then calculated. Note: in this way, correlations between parameters are fully taken into account in the errors on numerically determined parameters (including correlations with amplitudes), but for the errors on amplitudes only correlations with other amplitudes are taken into account.

2.5 Tolerance value estimation

The main criterion for convergence of a fit consists of comparing the relative change in the sum of the squared residuals between consecutive iterations to a set tolerance value. If this value is set too high, the fitting will halt before sufficient convergence is reached. Setting it too low, on the other hand, results in fits taking a very long time, although sufficient convergence was likely reached sooner. To obtain an accurate fit in a reasonable amount of time it is, therefore, important to set the tolerance to an appropriate value for any given PLE data. In order to provide some guidance in this decision, an equation is included in the GUI which suggests such a value based on a quick analysis of the data and fit, within 1-2 orders of magnitude. This equation was derived by extensive tests on simulated data in the following manner.

First, a single set of simulated data is fitted for 200 different instances of noise, with a very restrictive tolerance value $\tau = 10^{-9}$ to ensure proper convergence. The resulting fitted values for each parameter $\mathbb{P}_i$ are considered *fully converged* and used to determine the standard deviation σ_i. Subsequently, 30 of these instances are fitted again with a different convergence criterion: the fit is performed as usual but repeated multiple times, each time with a tolerance value one order of magnitude smaller than the last, starting from $\tau = 1$. When a tolerance value leads to fitted parameter values which are within $0.1\sigma_i$ of their *fully converged* values for all i, the fit is said to have converged and the current value of τ is then taken to be the appropriate tolerance for that set of data with that instance of noise.

The above scheme was repeated for different sequences of simulations. Within each sequence, a single aspect of the data was varied across these repetitions: number of chiral species, noise level, data resolution, data scale, ... The types of variations that have proved to be significant in these tests are (1) the number of chiral species N in the data and (2) the noise level L; the results for these are shown in Fig. S1.

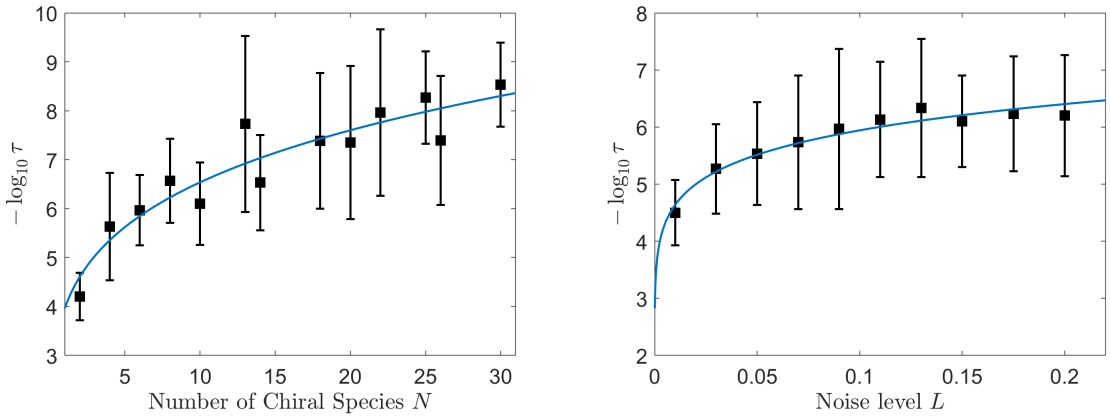


Fig. S1: Results from fits of simulated data: optimal tolerance value order of magnitude as a function of (*left*) number of chiral species N and (*right*) noise level L. In each figure, the black squares and error bars correspond to the mean and standard deviation across the 30 refitted instances of a particular simulation. The blue line shows the best fit, a power law in both cases.

The curves fitted to these results are combined in the GUI to suggest an appropriate order of magnitude for the tolerance value. This estimation function is given by

$$-\log_{10}\hat{\tau} = \alpha N^{e_1} L^{e_2}. \tag{S13}$$

The values and standard deviations for α, e_1 and e_2 are listed in Table S2. The accuracy of this estimator was tested with 15 more simulated samples and the average error on $-\log_{10}\hat{\tau}$ was found to be ± 0.57 with a standard deviation of 0.32. In other words, the ideal tolerance value for a given fit will most likely lie within one order of magnitude from the suggested value.

Fit Parameter	Value	Standard Deviation
α	5.84	0.82
e_1	0.23	0.08
e_2	0.11	0.02

Table S2: Fit parameters for the tolerance value order of magnitude $-\log_{10}\hat{\tau}$, as a function of N and L, in equation (S13).

In unsimulated data the noise level is estimated as follows. The noise itself is estimated by subtracting from the original data a smoothed version of it, created using a third-degree Savitzky-Golay filter in a 5 nm wide moving window, in each direction.[5] In other words, the data is fitted with a third-order polynomial in a region of 5 nm around each point to obtain the smoothed version. Subsequently, the average absolute value of the estimated noise is calculated. Assuming a fairly flat noise profile, this should correspond to half the maximum of the noise. This method is preferrable over simply taking the maximum of the estimated noise as that method is very sensitive to extreme outliers. The noise level itself is then estimated by dividing the thus-obtained maximum noise by the maximum of the data.

Due to the smoothing involved, the estimated noise will be less pronounced than the actual noise. Combined with the likelihood that the maximum of the data is higher than the underlying (real) intensity,

this means the estimate of the noise level will also be a lower bound. In fact, our simulations suggest the estimated level is typically between 80-100 % of the actual level, meaning the estimated tolerance value will be slightly smaller than the one given by (S13). Since this estimated value serves as a guide and shouldn't be taken as an absolute, such a slight underestimation of the noise level is not significant.

3 Additional Figures

Fig. S2 shows the 2D experimental PLE map and fit, together with 1D emission and excitation slices for a sample of HiPco SWCNTs after ATPE treatment for high (7,5) purity. The same graphs are shown in the main text for the parent HiPco sample (Fig. 2) and a different child sample ATPE treated for high (9,8) impurity (Fig. 4). A histogram as created by our GUI for each of these samples is moreover shown below in Fig. S3, indicating a typical use for this visualisation tool.

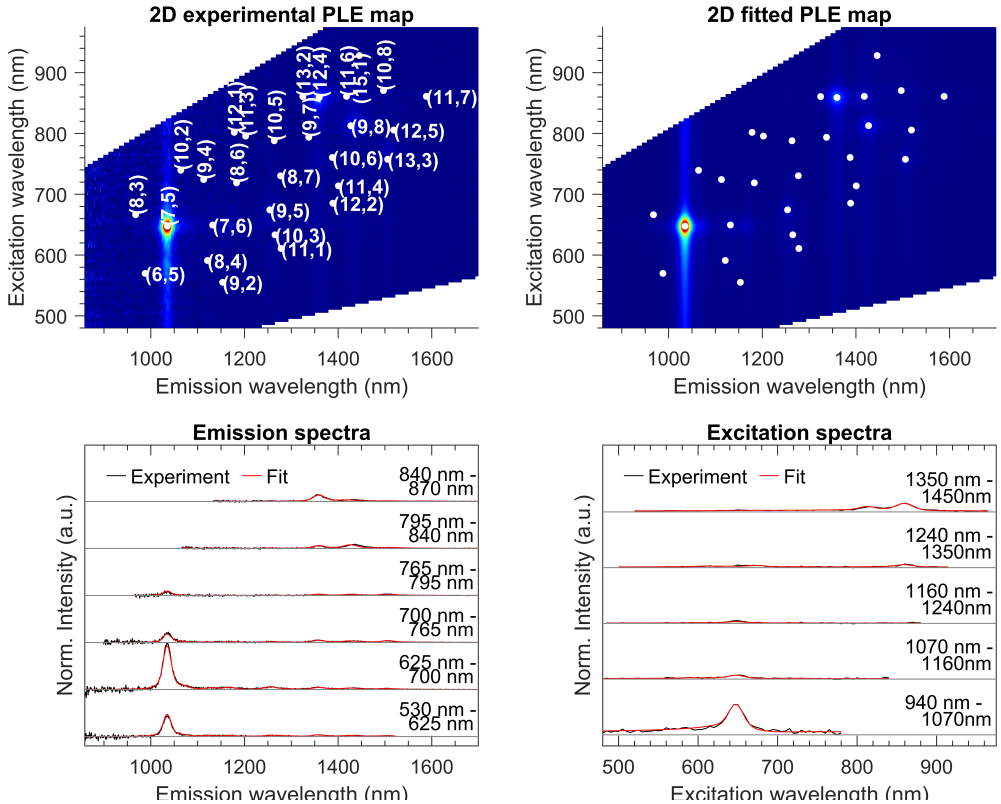
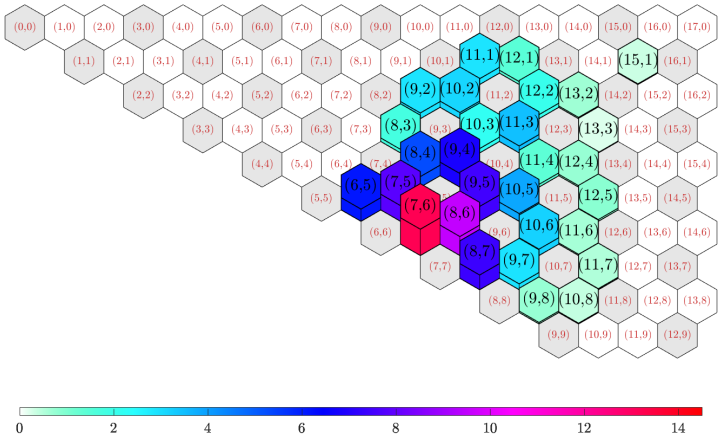
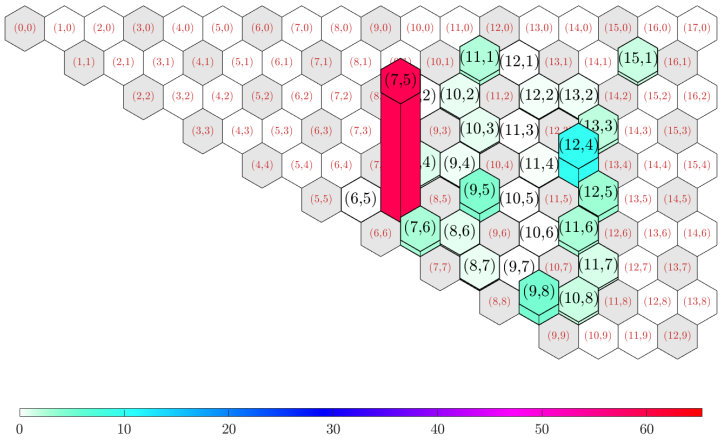


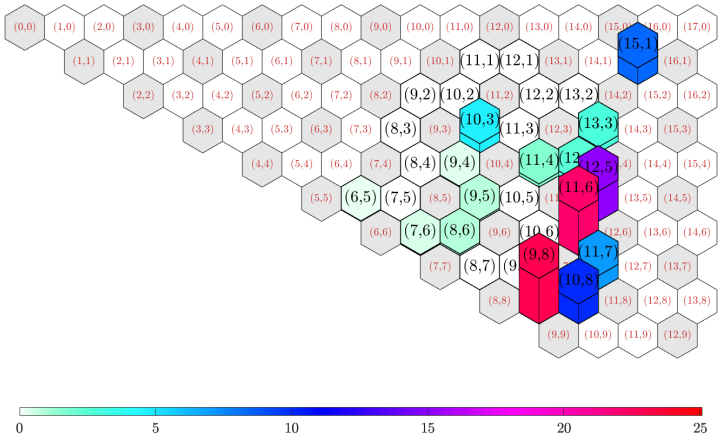
Fig. S2: PLE map and fit for an ATPE separated sample of HiPco SWCNTs with target chirality (7,5). (top left) Experimental 2D PLE map with chiral indices indicated. (top right) Fitted PLE map for this sample with fitted peak positions marked as white dots. (bottom left) Integrated emission spectra (black) and fits (red) for a few excitation intervals. (bottom right) Integrated excitation spectra (black) and fits (red) for a few emission intervals.



(a) Parent sample of empty HiPco SWCNTs.



(b) Child sample after ATPE treatment for high (7,5) purity.



(c) Child sample after ATPE treatment for high (9,8) purity.

Fig. S3: Histograms, analogous to Fig. 5 in the main text, generated by our GUI for the amplitudes extracted from fits of PLE maps for a parent sample and two of its child samples after different ATPE treatments. Intensities are given as a percentage of the total intensity.

References

[1] S. M. Bachilo, M. S. Strano, C. Kittrell, R. H. Hauge, R. E. Smalley and R. B. Weisman, *Science*, 2002, **298**, 2361–2366.

[2] S. Cambré and W. Wenseleers, *Angew. Chem. Int. Ed.*, 2011, **50**, 2764–2768.

[3] J. Campo, S. Cambré, B. Botka, J. Obrzut, W. Wenseleers and J. A. Fagan, *ACS Nano*, 2021, **15**, 2301–2317.

[4] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*, Society for Industrial and Applied Mathematics, 1974.

[5] A. Savitzky and M. J. E. Golay, *Analytical Chemistry*, 1964, **36**, 1627–1639.