

Supporting information:

Unveiling the Impact of Nanoparticle Size Dispersity on the Behavior of Polymer Nanocomposites

J. Javier Burgos-Mármol and Alessandro Patti*

School of Chemical Engineering and Analytical Science, The University of Manchester.

Sackville Street, M13 9PL, Manchester, UK

E-mail: alessandro.patti@manchester.ac.uk

S.1. Probability density function

To generate a polydisperse set of nanoparticles (NPs), a probability density function (pdf) is needed. One of the most common pdfs fitting NP size distribution is the Log-normal distribution, whose pdf is given in (Eq. S1):

$$f_{LN}(\sigma_n; \mu, \zeta) = \frac{1}{\zeta \sqrt{2\pi} \sigma_n} \frac{1}{\sigma_n} e^{-(\ln \sigma_n - \mu)^2 / 2\zeta^2}, \quad (\text{S1})$$

where σ_n is the NP's diameter, and the parameters μ and ζ are the mean and standard deviation, respectively, of the natural logarithm of σ_n . The average value of the diameter reads:

$$\langle \sigma_n \rangle = e^{\mu + \frac{\zeta^2}{2}}, \quad (\text{S2})$$

*To whom correspondence should be addressed

whereas the polydispersity index, Π_D , measuring the distribution of the NP's size, is defined as

$$\Pi_D \equiv \frac{\nu}{\langle \sigma_n \rangle^2} = e^{\zeta^2} - 1, \quad (\text{S3})$$

with $\nu = [\exp(\zeta^2) - 1][\exp(2\mu + \zeta^2)]$ the variance of the Log-normal distribution. The distributions used in this article to generate NPs are plotted in Fig. S1. The Log-normal moments read:

$$\langle \sigma_n^N \rangle = \langle \sigma_n \rangle^N (1 + \Pi_D)^{(N^2 - N)/2}, \quad \forall N \in \mathbb{R}. \quad (\text{S4})$$

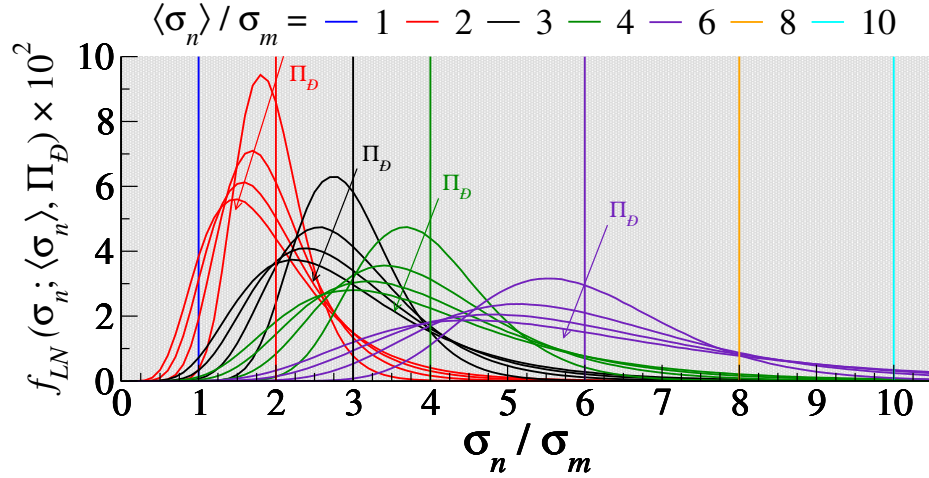


Figure S1: Log-normal probability density functions (Eq. S1) used in this work. Vertical lines are pdfs with $\Pi_D, \zeta \rightarrow 0$ and $\int_0^\infty d\sigma_n f_{LN}(\sigma_n; \mu, \zeta \rightarrow 0) = 1$. The other distributions have a non-zero standard deviation. In particular, their polydispersity indices are $\Pi_D = \{0.05, 0.10, 0.15, 0.20\}$.

To exclude from this distributions NPs with size unrealistically smaller than the polymer bead diameter, only NPs with $\sigma_n / \sigma_m \geq 1$ were selected. When an attempt to create a NP did not fulfil this condition, a new attempt was made. Mathematically, this process translates into the area below the cutoff diameter being removed and log-normally redistributed along the remaining part of the distribution (Fig. S2). Thus, the distribution used to generate the NPs reads:

$$f_{LNC}(\sigma_n; \mu, \zeta, d_c) = \frac{H(\sigma_n - d_c)}{\Omega} f_{LN}(\sigma_n; \mu, \zeta) = \frac{H(\sigma_n - d_c)}{\Omega} \frac{1}{\zeta} \frac{1}{\sqrt{2\pi} \sigma_n} e^{-\frac{(\ln \sigma_n - \mu)^2}{2\zeta^2}}, \quad (\text{S5})$$

where $d_c / \sigma_m = 1$ is the cutoff diameter, H is Heaviside's step function, and Ω the normalization

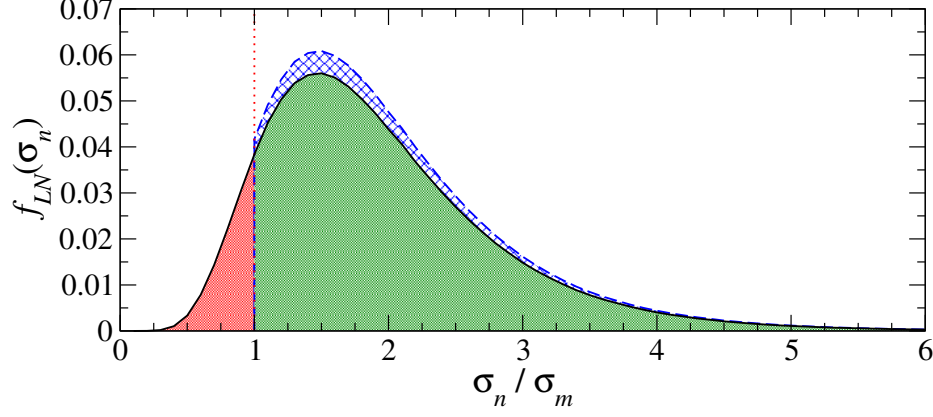


Figure S2: Log-normal probability density function (Eq. S1) with $\langle \sigma_n \rangle / \sigma_m = 2$ and $\Pi_D = 0.2$ (black continuous line), exhibiting the removed area (red), its redistribution (blue) and the remaining area of the original LN pdf (green). The modified pdf (Eq. S5) is also outlined (blue dashed line). The red dotted line stands at the cutoff diameter $d_c / \sigma_m = 1$.

constant. The normalization constant and the moments of the modified Log-normal distribution respectively read:

$$\Omega = \int_0^\infty d\sigma_n f_{LNc}(\sigma_n; \mu, \zeta, d_c) = \frac{1}{2} \operatorname{erfc} \left[\frac{\ln(d_c \sqrt{1 + \Pi_D} / \langle \sigma_n \rangle)}{\sqrt{2 \ln(1 + \Pi_D)}} \right], \quad (\text{S6})$$

$$\langle \sigma_n^N \rangle_{d_c} = \frac{\langle \sigma_n \rangle^N (1 + \Pi_D)^{(N^2 - N)/2}}{2\Omega} \operatorname{erfc} \left[\frac{\ln(d_c (1 + \Pi_D)^{1/2 - N} / \langle \sigma_n \rangle)}{\sqrt{2 \ln(1 + \Pi_D)}} \right], \quad \forall N \in \mathbb{R}, \quad (\text{S7})$$

where $\operatorname{erfc}(x)$ is the complementary error function, and $\langle \dots \rangle_{d_c}$ means average over (Eq. S5). The value of Ω , corresponding the red (or blue) area in Fig. S2, is plotted in Fig. S3. For most of the systems studied in this article $\Omega < 1\%$, which, to a good approximation, makes the NP size distribution in these systems virtually Log-normal. Nevertheless, the few systems with larger values of Ω may show non-negligible discrepancies in their predictions. As a consequence, the use of Eq. (S5) is imperative in order to validate our simulation data.

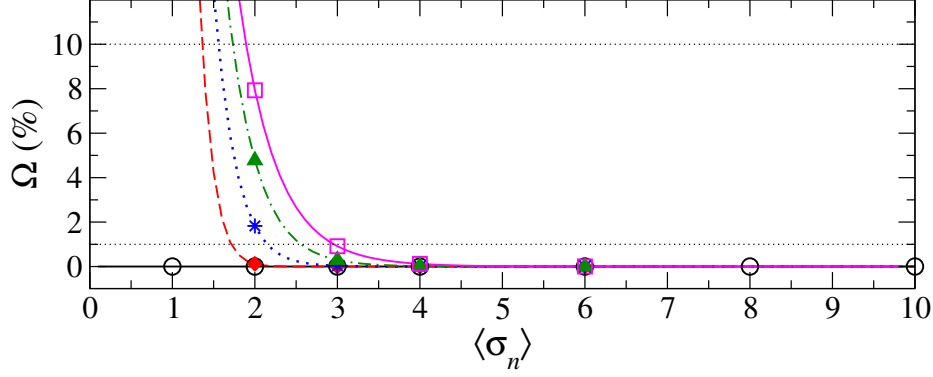


Figure S3: Normalization constant of (Eq. S2). Lines are the curves at constant $\Pi_D = 0.00$ (continuous black), $\Pi_D = 0.05$ (dashed red), $\Pi_D = 0.10$ (dotted blue), $\Pi_D = 0.15$ (dashed dotted green), and $\Pi_D = 0.20$ (continuous magenta). Symbols correspond to the systems studied in this article.

S.2. Simulations set up

The model of NPs employed in this work, preventing the overlap of the NP cores either with each other or with the polymer chain beads, together with the NP size dispersity, may hamper or delay the process of equilibration. In order to overcome this problem, the dispersion of NPs inside the simulation box was addressed in first place. To this end, the initial configuration consisted of polymer chains sequentially arranged as in a lattice network at an end of the box, see Fig. S4, and separated from the NPs. The simulation box initially had a tetragonal geometry, with dimensions $L_{box} \times L_{box} \times L_{z0}$, where L_{box} is the length of the final cubic box side, and $L_{z0} > L_{box}$ the initial length of the elongated side. In particular, $L_{box} = [N_c L_c \rho_m^{-1} (1 - \phi_n)^{-1}]^{(1/3)}$, where N_c is the number of polymer chains, $L_c = 10$ the number of beads constituting the chain, $\rho_m / \sigma_m^{-3} = 0.7$ the target polymer chain bead number density, and ϕ_n the target NP volume fraction. To achieve the final density, we first allowed the NPs to perform Monte Carlo (MC) moves, according to the Metropolis algorithm and the Boltzmann condition,¹ while keeping the polymer chains frozen. The NPs were allowed to move to that region of the simulation box not occupied by the polymer chains. For every single MC cycle, one attempt to reduce L_z by a random length between 0 and $0.1\sigma_m$ was made. If accepted, the positions of the NPs were rescaled by the same factor inside the permitted region. The move was accepted if and only if no violation of the minimum allowed NP-NP and NP-monomer distances occurred, but did not follow the Boltzmann conditions, as it would in an

NPT ensemble. Typically, the system needed $\sim 10^2 - 10^3$ MC cycles before the box reached its final dimensions. An additional amount of 500 cycles were run to relax the NPs. Once a perfect cubic geometry was obtained, the whole system was equilibrated by running Molecular Dynamics (MD) simulations in LAMMPS,² with chains and NPs moving freely in the conditions described in the article. A minimum amount of 6×10^6 , and up to 10^7 , MD steps were needed to relax the energy to a steady value within statistical fluctuations.

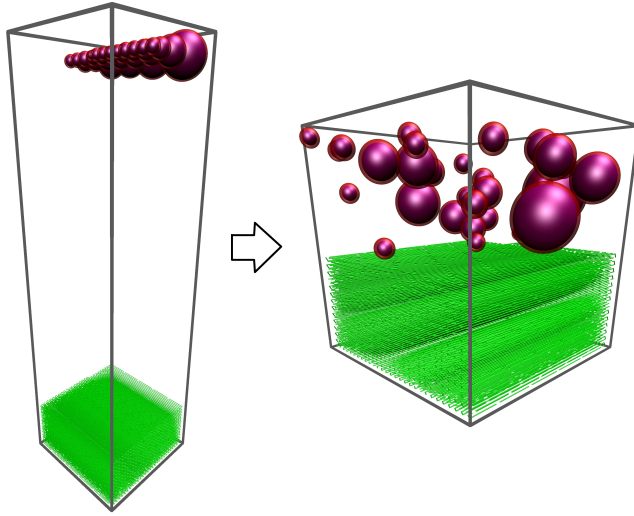


Figure S4: Representative snapshots of the process of equilibration of a polymer melt incorporating 30 polydisperse ($\langle \sigma_n \rangle / \sigma_m = 6$, $\Pi_D = 0.20$) NPs, and 7798 polymer chains. The hard core and soft shell of the NPs are shown in purple and semi transparent red, respectively, while the polymer chains are displayed in green and reduced in size for clarity. In this particular case, an elongated simulation box of dimensions $V_{box,0} / \sigma_m^3 = 48.95 \times 48.95 \times 175.81$ (left frame) was set containing a crystalline ordered polymer at one end and an arrangement of ordered NPs at the other. A MC simulation were then performed, allowing NP moves and box resizing, while polymer chains remained in their initial positions. When the target volume was obtained, namely $V_{box} / \sigma_m^3 = 48.95 \times 48.95 \times 48.95$ (center frame), an MD simulation was run, allowing every component to move freely until the equilibrium configuration is reached (right frame). This figure has been generated with the Visual Molecular Dynamics (VMD) software.³

S.4. Ould-Kaddour-Levesque's average diffusivity for $d_c/\sigma_m < 1$

The average diffusivity obtained from Ould-Kaddour and Levesque's model, must be calculated separately for the cases $d_c/\sigma_m < 1$ and $d_c/\sigma_m \geq 1$, the latter being the case discussed in the article.

For the sake of completeness, we show here the average value left out of the main publication:

$$\langle D_n^{OL} \rangle_{d_c < 1} = \frac{1}{\Omega} \left\{ \langle D_n^{OL} \rangle + \frac{k_B T}{6\pi\eta_0} \sum_{N=1}^{\infty} (-1)^N \frac{\langle \sigma_n \rangle^{N-1}}{\sigma_m^N} (1 + \Pi_D)^{N(N-3)/2+1} \operatorname{erfc} \left[\frac{\ln \left(\frac{\langle \sigma_n \rangle}{d_c} (1 + \Pi_D)^{N-3/2} \right)}{\sqrt{2 \ln(1 + \Pi_D)}} \right] \right\}, \quad (\text{S8})$$

where

$$\begin{aligned} \langle D_n^{OL} \rangle = & \frac{k_B T (1 + \Pi_D)}{6\pi\eta_0 \langle \sigma_n \rangle} \left\{ \operatorname{erfc} \left(\frac{\ln \left(\frac{\sigma_m}{\langle \sigma_n \rangle} (1 + \Pi_D)^{3/2} \right)}{\sqrt{2 \ln(1 + \Pi_D)}} \right) + \right. \\ & + \sum_{N=1}^{\infty} (-1)^N \left[- \left(\frac{\langle \sigma_n \rangle}{\sigma_m} \right)^N (1 + \Pi_D)^{N(N-3)/2} \operatorname{erfc} \left(\frac{\ln \left(\frac{\langle \sigma_n \rangle}{\sigma_m} (1 + \Pi_D)^{N-3/2} \right)}{\sqrt{2 \ln(1 + \Pi_D)}} \right) + \right. \\ & \left. \left. + \left(\frac{\sigma_m}{\langle \sigma_n \rangle} \right)^N (1 + \Pi_D)^{N(N+3)/2} \operatorname{erfc} \left(\frac{\ln \left(\frac{\sigma_m}{\langle \sigma_n \rangle} (1 + \Pi_D)^{N+3/2} \right)}{\sqrt{2 \ln(1 + \Pi_D)}} \right) \right] \right\}, \quad (\text{S9}) \end{aligned}$$

and k_B is Boltzmann's constant, T is the absolute temperature, and η_0 the viscosity of the pure polymer.

S.5. Additional results on NP diffusivity

The models by Yamamoto and Schweizer, and Ould-Kaddour and Levesque, are respectively compared to our results in Figs. S5 and S6. While Yamamoto and Schweizer's model overestimates the diffusivity values when $\sigma_n < 2R_g$, Ould-Kaddour and Levesque's underpredict them.

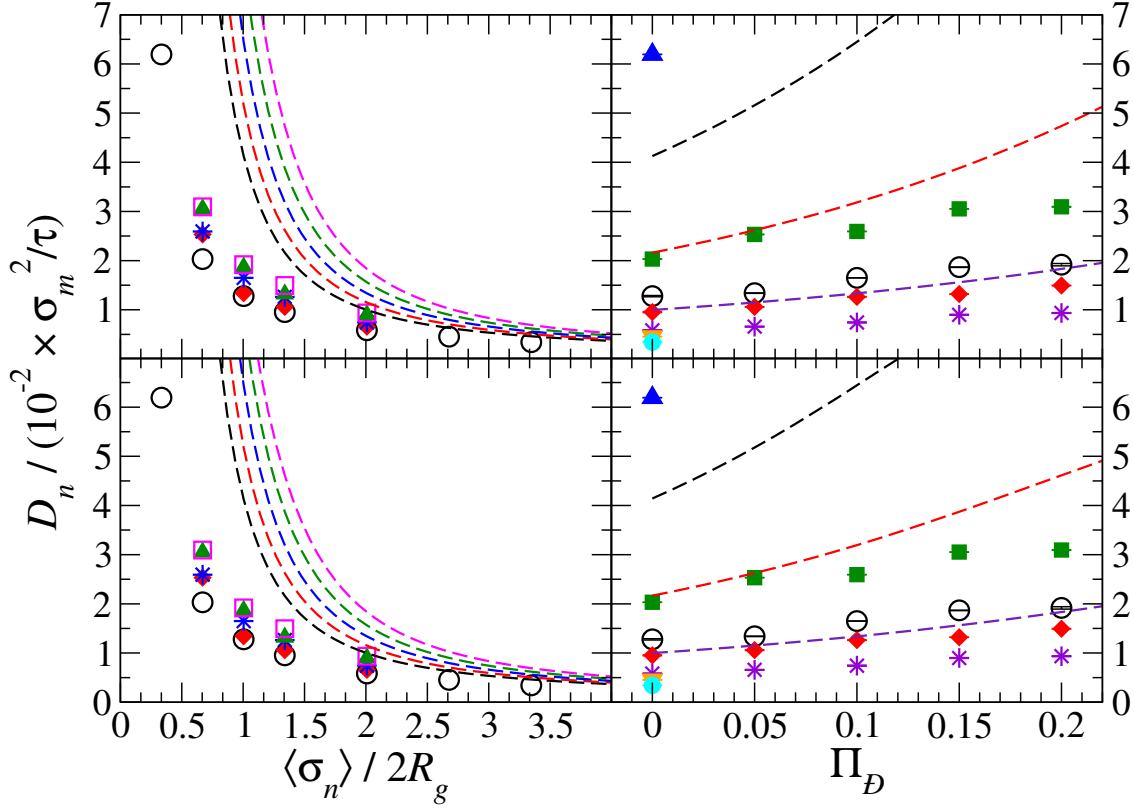


Figure S5: Average diffusion coefficients of NPs in melts of unentangled polymer chains as a function of the average NP diameter (a, c) and polydispersity index (b, d). Symbols refer to simulation results, whereas dashed lines to the theoretical predictions of the Yamamoto and Schweizer's model within a Log-normal (a, b) and modified Log-normal (c, d) size distribution. Left frames: $\Pi_D = 0.00$ (○), 0.05 (◆), 0.10 (*), 0.15 (▲), and 0.20 (◻). Right frames: $\langle \sigma_n \rangle / 2R_g = 1/3$ (▲), 2/3 (■), 1 (○), 4/3 (◆), 2 (*), 8/3 (▼), and 10/3 (●). The range of colors used for the dashed lines follows the same description as the symbols. Error bars are smaller than the symbol size and might not be visible.

References

- (1) Frenkel, D.; Smit, B. *Understanding Molecular Simulation: From Algorithms to Applications*, 2nd ed., Academic Press, 2002

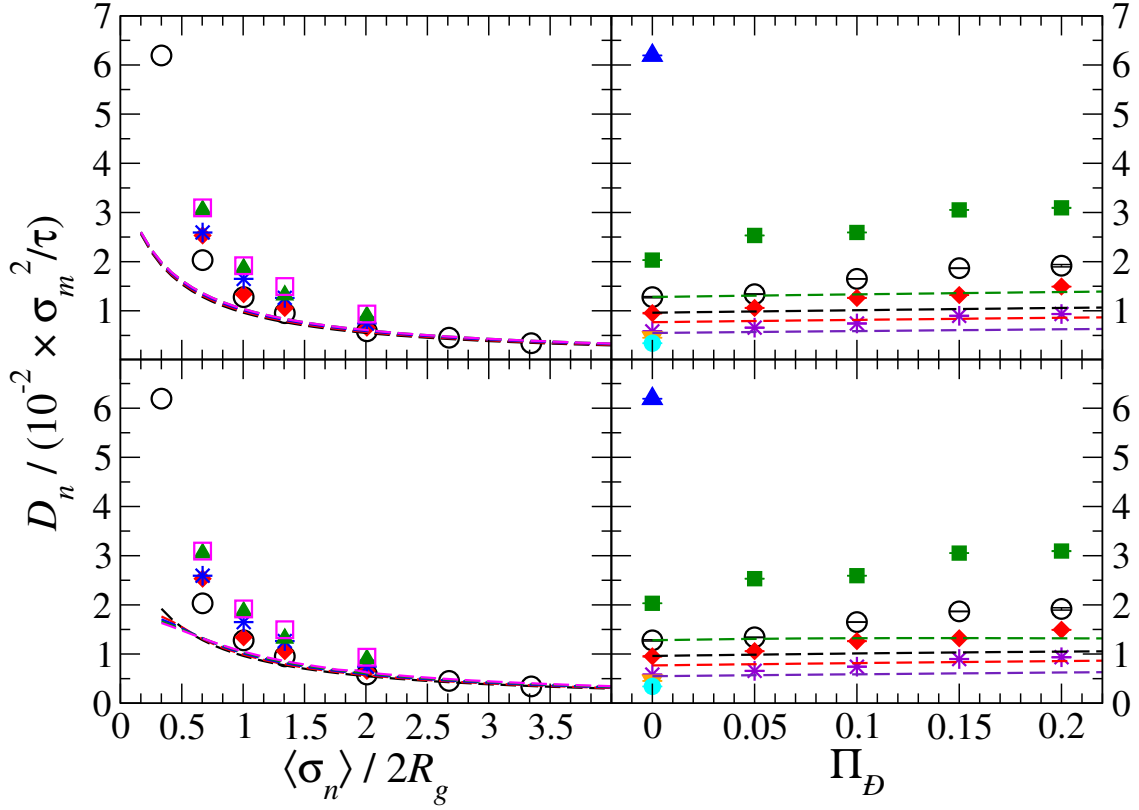


Figure S6: Average diffusion coefficients of NPs in melts of unentangled polymer chains as a function of the average NP diameter (a, c) and polydispersity index (b, d). Symbols refer to simulation results, whereas dashed lines to the theoretical predictions of the Ould-Kaddour and Levesque's model within a Log-normal (a, b) and modified Log-normal (c, d) size distribution. Left frames: $\Pi_D = 0.00$ ($\circ$), 0.05 ($\blacklozenge$), 0.10 ($\ast$), 0.15 ($\blacktriangle$), and 0.20 ($\square$). Right frames: $\langle\sigma_n\rangle/2R_g = 1/3$ ($\blacktriangle$), $2/3$ ($\blacksquare$), 1 ($\circ$), $4/3$ ($\blacklozenge$), 2 ($\ast$), $8/3$ ($\blacktriangledown$), and $10/3$ ($\bullet$). The range of colors used for the dashed lines follows the same description as the symbols. Error bars are smaller than the symbol size and might not be visible. Mathematica was used to compute the theoretical curves.⁴

- (2) *LAMMPS - Large-scale Atomic/Molecular Massively Parallel Simulator*, versions Feb 14 and May 15; Plimpton, S. *J. Comput. Phys.* **1995**, *117*, 1-19
- (3) Humphrey, W.; Dalke, A.; Schulten, K. *J. Molec. Graphics* **1996**, *14*, 33-38
- (4) *Mathematica*, version 10.3; software for technical computation; Wolfram Research: Champaign, IL 2015