Supplementary Material for the article titled "Morphology of Compound Viscoelastic Drops in Extensional Flows"

Malay Vyas[1] and Uddipta Ghosh[1]

[1]*Department of Mechanical Engineering, Indian Institute of Technology Gandhinagar, Palaj - 382055, Gujarat, India*

The purpose of this supplementary material is to provide additional details on various aspects discussed in the main article. To this end, in §S1, we lay out the detailed implementation of the viscoelastic constitutive relation and its coupling with the ternary phase field model in COMSOL® Multiphysics (a commercial software package) - this complements §4.1 and §4.2 in the main article. §S2 provides results for the grid and time-step independence of our numerical simulations which complements §4.2 of the main manuscript. Some additional results, particularly on the transient variations in the deformation factor along with another set of comparison between asymptotic and numerical simulations have been carried out in §S3 - this complements §5 of the main manuscript. Finally, description of the Maple™files along with brief guidelines for their use towards evaluating the various constants appearing as part of the $O(De)$ and $O(De^2)$ asymptotic solutions have been incorporated in §S4.

S1 Numerical implementation in COMSOL Multiphysics® Environment

Here, we illustrate in detail the combined implementation of the ternary phase field model, creeping-flow, and the Giesekus constitutive relationin the COMSOL Multiphysics 5.6 Environment. The implementation is done using the following built-in modules in COMSOL: (i) the Laminar Flow Module for Stokes flow, (ii) the Ternary Phase Field Module to handle the three distinct phases, and (iii) the PDE Module for implementing the Giesekus constitutive model, along with the Viscoelastic Flow Module for enforcing the boundary conditions for the polymeric stresses. In the sub-sections ahead, we provide a step-by-step description of the implementation focusing on each module at a time.

S1.1 The Laminar Flow Module - Creeping Flow

The Laminar Flow Module, with inertial advective terms neglected, is used setting to simulate the low Reynolds number flow problem under consideration (see Eq. (2a) in the manuscript). This module solves the mass and momentum conservation equations which take the following form (throughout this section we report the equations in the same form as they appear in the COMSOL Multiphysics® environment[1], for ease of future reference):

$$\rho \boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 \tag{S1a}$$

$$\rho \frac{\partial \boldsymbol{u}}{\partial t} = \boldsymbol{\nabla} \cdot [-p\boldsymbol{I} + \boldsymbol{K}] + \boldsymbol{F} \tag{S1b}$$

Here, the dimensional quantities will be replaced by the non-dimensional numbers and quantities as per the scheme discussed in the manuscript. The total stress $\boldsymbol{\tau}$ of the Giesekus constitution is split into a viscous part and an elastic part as $\boldsymbol{\tau} = \boldsymbol{\tau}_s + \boldsymbol{\tau}_p$, where the viscous part is already implemented in the term $\boldsymbol{K} = \mu_s(\boldsymbol{\nabla u} + \boldsymbol{\nabla u}^T)$ in Eq. (S1b) - this is identical to the Newtonian deviatoric stress. The polymeric stress contribution $\boldsymbol{\nabla} \cdot \boldsymbol{\tau}_p$ is implemented through the body force term $\boldsymbol{F}$ appearing in Eq. (S1b). Further details have been provided in the subsection ahead in §S1.3. The splitting of the viscous stresses as performed above is known as Elastic-Viscous Stress Splitting (EVSS)[2].

The creeping-flow problem is complemented with initial velocity field set to $\mathbf{0}$, while the far-field imposed velocities are prescribed at the top and the right boundaries of the solution domain - refer to Figure S1 for a schematic depiction of the numerical simulation domain (this is same as Fig. 2 in the main article). A symmetry condition is imposed on the bottom boundary. For a 2D axisymmetric geometry, the left boundary is enforced as the axis of symmetry. The value of pressure has also been provided at a suitable point.

S1.2 The Ternary Phase Field Module

The Ternary phase field module follows the implementation described by Boyer et al.[3] coupled with the Laminar Flow Module (§S1.1). The reader may refer to the original papers[3, 4] and documentation provided by COMSOL[5] for further

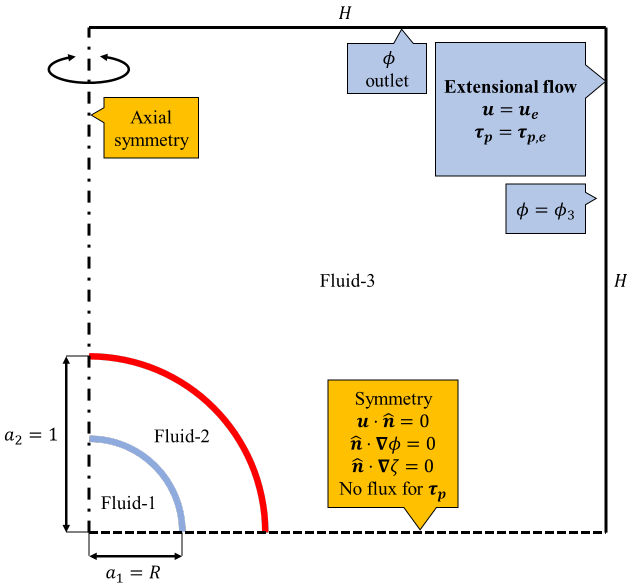

Figure S1: Schematic of the numerical simulations domain is shown. It only consists of a quarter of the entire flow domain because of the axisymmetric nature of the problem. The axis of symmetry (z axis), the $z = 0$ plane (plane of symmetry) and the domain boundaries have been depicted. The domain size is $H \times H$. Initial distribution of the three phases have also been shown with the core (fluid-1) having radius R, the shell with radius 1 at $t = 0$. The conditions satisfied on the four boundaries by all relevant variables have also been shown in the schematic.

details of the ternary phase field model. This module solves modified Cahn-Hilliard type equations [6] for phases A and B, given by (also see Eq. (18) in the main article for meaning of the various symbols):

$$\frac{\partial \phi_i}{\partial t} + \boldsymbol{\nabla} \cdot (\mathbf{u}\,\phi_i) = \boldsymbol{\nabla} \cdot \left(\frac{M_0}{\Sigma_i} \boldsymbol{\nabla} \eta_i \right) \tag{S2a}$$

$$\eta_i = \frac{4\Sigma_T}{\varepsilon} \sum_{i \neq j} \left(\frac{1}{\Sigma_j} \left(\partial_i F(\Phi) - \partial_j F(\Phi) \right) \right) - \frac{3}{4} \varepsilon \Sigma_i \nabla^2 \phi_i, \;\; i \equiv A,\, B \tag{S2b}$$

On the other hand, presence of phase C is calculated using $\phi_C = 1 - (\phi_A + \phi_B)$. We emphasize that the above equations have been written using the notations given in COMSOL. The ternary phase field module requires the specification of a few important properties which, in our case, will boil down to the Cahn number as the non-dimensional interfacial thickness (ε) and the non-dimensional mobility tuning parameter - see Eq. (19) in the main article. The additional free bulk energy term Λ appearing in the expression of $F = \sigma_{ij} \sum_{i \neq j} \phi_i \phi_j + \phi_A \phi_b \phi_c \sum_{k=A,B,C} (\Sigma_k \phi_k) + \Lambda \phi_A^2 \phi_B^2 \phi_C^2$ is set to zero, while the dimensionless surface tension is incorporated using the σ_{ij} terms. Definitions of $\Sigma_i = \sigma_{ij} + \sigma_{ik} - \sigma_{jk}$ and $\frac{1}{\Sigma_T} = \frac{1}{\Sigma_A} + \frac{1}{\Sigma_B} + \frac{1}{\Sigma_C}$ are as described in the literature. The initial conditions are then used to set up the three phases with $\phi_A = 1, \phi_B = 0$ for the core, $\phi_A = 0, \phi_B = 1$ for the shell, and $\phi_A = 0, \phi_B = 0$, i.e. $\phi_C = 1$ for the outermost phase. The axis of symmetry and the symmetry boundary conditions remain same as discussed in §S1.1. In addition, at the inlet, the condition $\phi_{\text{inlet}} = \phi_C$ is imposed for the right boundary and a similar outlet condition is also specified at the top boundary to represent the physical nature of the problem.

A 'multiphysics node' is needed to couple the one-fluid formulation which combines the velocity and pressure calculations from the laminar flow module and the phase tracking through the ternary phase-field module. The "*Three-Phase Flow, Phase-Field*" node, which provides a default coupling between the two modules, is used where the (dimensionaless) fluid properties are specified (see the subsection ahead for this). We would like to clarify that the dynamic viscosities included in our implementation represent the solvent's contribution to the Giesekus constitutive model.

S1.3 The Partial Differential Equation (PDE) Module for Viscoelasticity

The elastic stresses are incorporated in Eq. (S1b) using the body force term $\boldsymbol{F}$. The various fluid properties (such as the relaxation time, polymer and solvent viscosity, etc.) are defined using Eq. (20) in the manuscript which may be explicitly written as:

$$D = De(\Lambda_1 \phi_A + \Lambda_2 \phi_B + \Lambda_3 \phi_C); \quad \alpha = \alpha_1 \phi_A + \alpha_2 \phi_B + \alpha_3 \phi_C; \quad \upsilon_s = \upsilon_{s,1} \phi_A + \upsilon_{s,2} \phi_B + \upsilon_{s,3} \phi_C \tag{S3a}$$

$$\upsilon_p = 1 - \upsilon_s; \quad \xi = \xi_1 \phi_A + \phi_B + \xi_3 \phi_C; \quad Z = \upsilon_p \xi \;\; \text{and} \;\; A = \frac{\alpha D}{Z} \tag{S3b}$$

We use the coefficient form of PDE with four dependent variables $\tau_{p,rr}, \tau_{p,rz}, \tau_{p,\theta\theta}, \tau_{p,zz}$ (defined in cylindrical coordinates to take advantage of axisymmetry - see Fig. S1) to account for the four non-zero components of the symmetric

polymeric stress tensor. In the rest of this section, the radial direction in cylindrical coordinate has been denoted as r (as per COMSOL notations) instead of ρ used in the main article. The Giesekus constitutive relation [7] (also see Eq. (19c) in the main article) is then used to write the following component-wise equations relating the polymeric stresses with the flow field and the relevant fluid properties:

$$D\left(\frac{\partial \tau_{p,rr}}{\partial t}+u\frac{\partial \tau_{p,rr}}{\partial r}+w\frac{\partial \tau_{p,rr}}{\partial z}\right)+\tau_{p,rr}=2Z\frac{\partial u}{\partial r}+2D\left(\tau_{p,rr}\frac{\partial u}{\partial r}+\tau_{p,rz}\frac{\partial u}{\partial z}\right)-A\left(\tau_{p,rr}^2+\tau_{p,rz}^2\right) \tag{S4a}$$

$$D\left(\frac{\partial \tau_{p,rz}}{\partial t}+u\frac{\partial \tau_{p,rz}}{\partial r}+w\frac{\partial \tau_{p,rz}}{\partial z}\right)+\tau_{p,rz}=Z\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial r}\right)-A\left(\tau_{p,rz}(\tau_{p,rr}+\tau_{p,zz})\right)+D\left(\tau_{p,rr}\frac{\partial w}{\partial r}+\tau_{p,rz}\left(\frac{\partial w}{\partial z}+\frac{\partial u}{\partial r}\right)+\tau_{p,zz}\frac{\partial u}{\partial z}\right) \tag{S4b}$$

$$D\left(\frac{\partial \tau_{p,\theta\theta}}{\partial t}+u\frac{\partial \tau_{p,\theta\theta}}{\partial r}+w\frac{\partial \tau_{p,\theta\theta}}{\partial z}\right)+\tau_{p,\theta\theta}=2Z\frac{u}{r}+2D\left(\tau_{p,\theta\theta}\frac{u}{r}\right)-A\left(\tau_{p,\theta\theta}^2\right) \tag{S4c}$$

$$D\left(\frac{\partial \tau_{p,zz}}{\partial t}+u\frac{\partial \tau_{p,zz}}{\partial r}+w\frac{\partial \tau_{p,zz}}{\partial z}\right)+\tau_{p,zz}=2Z\frac{\partial w}{\partial z}+2D\left(\tau_{p,rz}\frac{\partial w}{\partial r}+\tau_{p,zz}\frac{\partial w}{\partial z}\right)-A\left(\tau_{p,rz}^2+\tau_{p,zz}^2\right) \tag{S4d}$$

In Eq. (S4), $\boldsymbol{u}=u\hat{\mathbf{e}}_r+w\hat{\mathbf{e}}_z$ is the velocity field, with u and w respectively being the r and the z components. The above system of non-linear partial differential equations may be modeled using the coefficient form PDE available in the PDE module in COMSOL 5.6. The general form of this PDE is as follows (using COMSOL notation) for a variable vector $\boldsymbol{\chi}$ which itself may contain a given set of dependent variables:

$$e_a\frac{\partial^2\boldsymbol{\chi}}{\partial t^2}+d_a\frac{\partial\boldsymbol{\chi}}{\partial t}+\boldsymbol{\nabla}\cdot(-c\boldsymbol{\nabla}\boldsymbol{\chi}-\boldsymbol{\alpha}\boldsymbol{\chi}+\boldsymbol{\gamma})+\boldsymbol{\beta}\cdot\boldsymbol{\nabla}\boldsymbol{\chi}+a\boldsymbol{\chi}=\boldsymbol{f} \tag{S5}$$

With $\boldsymbol{\nabla}=\left[\frac{\partial}{\partial r},\frac{\partial}{\partial z}\right]$ being the gradient operator for the axisymmetric case. In the present scenario, we may choose,

$$\boldsymbol{\chi}=[\tau_{p,rr},\,\tau_{p,rz},\,\tau_{p,\theta\theta},\,\tau_{p,zz}]^T \tag{S6}$$

Comparing Eq. (S5) and Eq. (S4), it is evident that $d_a=diag(D)$, $a=diag(1)=I$, $\beta=diag\left(\begin{bmatrix}\beta_r=uD\\ \beta_z=wD\end{bmatrix}\right)$, and $\boldsymbol{f}$ may be defined as the right hand sides of Eq. (S4). The rest of the coefficients, namely, e_a, c, α, γ are identically set to zero to recover Eqs. (S4a)-(S4d). Here, '$diag(\Xi)$' represents a square matrix whose diagonal entries are the elements of Ξ. Note that the velocity components and the order parameters which are estimated using other modules appear implicitly in these equations and maintain the coupling between the three modules.

The set of PDEs defined above represent the elastic contributions from the Giesekus model. The flow is started from rest with zero elastic stresses at $t=0$ (initial condition), and zero normal gradient condition on the plane of symmetry (bottom boundary); the boundary conditions become fixed in COMSOL upon specifying the their types. The axis of symmetry (the left boundary) is treated implicitly according to the geometry in a way similar to the previous subsections. The top and the right boundaries, where the velocities are specified, need additional treatment. To determine the stress field pertaining to the far field, essentially represented by the right and the top boundaries (see Fig. S1), we first perform a steady-state simulation using the in-built Viscoelastic Flow Module with the properties of the suspending fluid (i.e., fluid-3 or, C), and the imposed velocities (see Eq. (1) in the main article). We assume that the far-field conditions are not disturbed by the presence of the drop, and hence the steady state solutions for the elastic stresses obtained from the simulations stated above are used as Dirichlet conditions on the right and the top boundaries to specify the various components of $\boldsymbol{\tau}_p$ there.

The solution of the system of PDEs described above (coupled with the variables from the other modules) yields the unknown polymeric stress components for the Giesekus constitutive model at each time step. Through the two-way coupling between the PDE module and the remaining two modules, the velocity field and the distribution of order parameters are also estimated accurately using information about the elastic stresses. To accomplish the latter, the elastic (polymeric) stresses are implemented in the momentum conservation equation (Eq. (S1b)) by appropriately defining the components of the body force term $\boldsymbol{F}$ as follows: $\boldsymbol{F}=\boldsymbol{\nabla}\cdot\boldsymbol{\tau}_p=F_r\hat{\mathbf{e}}_\mathbf{r}+F_z\hat{\mathbf{e}}_\mathbf{z}$, with F_r and F_z defined as:

$$F_r=\frac{\partial \tau_{p,rr}}{\partial r}+\frac{\tau_{p,rr}}{r}+\frac{\partial \tau_{p,rz}}{\partial z}-\frac{\tau_{p,\theta\theta}}{r} \tag{S7a}$$

$$F_z=\frac{\partial \tau_{p,rz}}{\partial r}+\frac{\tau_{p,rz}}{r}+\frac{\partial \tau_{p,zz}}{\partial z} \tag{S7b}$$

This force vector F is then prescribed as a volumetric force term for the entire flow domain in the Laminar flow module, representing elastic part of the constitutive relationship.

Summarily, the laminar flow module solves for the velocity field in combination with the ternary phase field module which accounts for the presence of three distinct phases. The elastic stresses are solved using a system of partial differential equations implemented in the PDE module. These stresses are then embedded within the momentum conservation

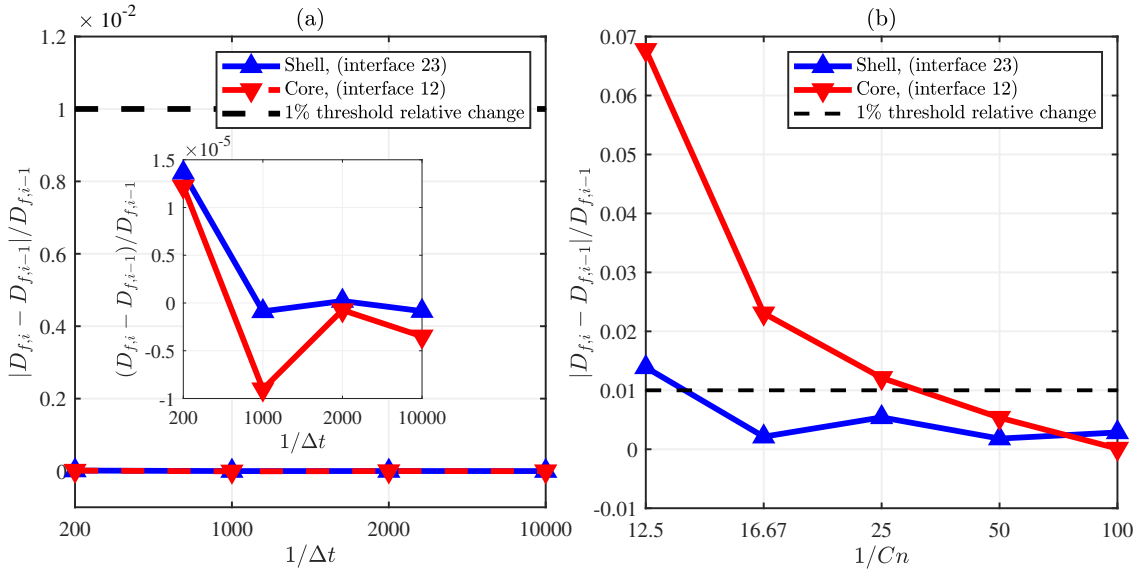

Figure S2: (a) Time step independence of the numerical simulations is shown by plotting the relative change in the steady-state deformation factors for the shell and the core against successive time-step refinements. The inset shows the same variation albeit with a smaller y-span. (b) Cahn number (Cn)-cum-grid size independence of the numerical simulations is shown by plotting the relative change in the steady-state deformation factor (for the shell and the core) against $1/Cn$ ranging between $1/0.08$ and $1/0.01$. Other relevant parameters are: $Ca = 0.05, De = 0.1, R = 0.5, \upsilon_1 = \upsilon_2 = 0.5, \upsilon_3 = 1, \xi_1 = \xi_3 = 1, \Lambda_1 = 1, \alpha_1 = \alpha_2 = \alpha_3 = 0, \kappa = 1$.

equation of the laminar flow module through a volumetric force term. Dirichlet boundary conditions for the elastic stresses are separately computed making use of an in-built stationary viscoelastic flow solver. The three transient modules (Laminar, ternary phase-field, PDE) are then solved simultaneously using a segregated solver, wherein an in-built PARDISO scheme is deployed.

S2 Grid independence and time-step independence studies

Figure S2(a) illustrates the evolution of the results for the shell and core deformations with changes in time step sizes from $\Delta t = 0.01$ down to 0.0001. We plot the relative change in the steady state deformation with each successive time step refinement, defined as, $\frac{|D_{f,i} - D_{f,i-1}|}{D_{f,i-1}}$, where $D_{f,i}$ indicates the steady state deformation for the i-th choice of the time-step size, as a function of the inverse time step, $1/\Delta t$. In all instances, the simulations are initiated with relatively small time steps $\Delta t_{\text{initial}} = 0.001\Delta t$ in order to ensure that the temporal variations are properly resolved and the startup transience is captured accurately (see Fig. S4). The step size is gradually increased thereafter to a pre-specified Δt with a growth rate of 1.5 (per time step), after which the Δt is maintained constant. It is evident that for $\Delta t < 0.01$, any subsequent refinement in the time step does not lead to any noticable change in the steady-state deformation, and the relative change is always well below 1%, which may be taken as a threshold level (a dotted line is included to represent a 1% relative change). However, it was noted that in certain scenarios, especially at relatively large De values, simulations with relatively a high time step size ($\Delta t \approx 0.01$) may result in convergence issues. Therefore, in this work, a time step of $\Delta t = 0.001$ is chosen for all numerical simulations. The inset figure shows the same relative variation as in the main figure with a tighter y axis limit, and establishes that the overall change is $O(10^{-5})$ - well within our threshold.

In phase-field formalism, the mesh size is generally chosen depending on the Cahn number (Cn), an indicator of the typical interface thickness. While mesh-size independence is mandatory, it is usually also required to achieve Cn-independence, i.e., as $Cn \to 0$, the numerical results should approach the sharp interface limit asymptotically [8]. Since the mesh size (Δx) depends on Cn, it is sufficient to test for Cn-independence, which will also imply mesh-independence. Figure S2(b) exhibits the results of the Cahn number independence tests, by plotting the relative change in the steady state deformation with each successive refinement in Cn defined as $\frac{|D_{f,i} - D_{f,i-1}|}{D_{f,i-1}}$ as a function of $1/Cn$ (here, $D_{f,i}$ is the steady-state deformation for the i-th choice of Cn). In all cases, we have taken $\Delta x = Cn/5$ (mesh size), while a time step size of $\Delta t = 0.001$ was chosen. The Cahn number was varied from 0.08 down to 0.01. It is evident that for $Cn \leq 0.02$, the relative change in the steady state deformation falls below 1% (threshold level shown by a dotted line) for both the shell and the core. In fact, the deformation of the core seems to show a greater degree of sensitivity towards the choice of Cn as compared to the shell. In the main article, all results have been reported for $Cn = 0.02$ and a minimum grid size of $\Delta x_{\text{min}} = Cn/5$ (unless otherwise mentioned), in an effort to maintain a permissible computational time without compromising on accuracy.

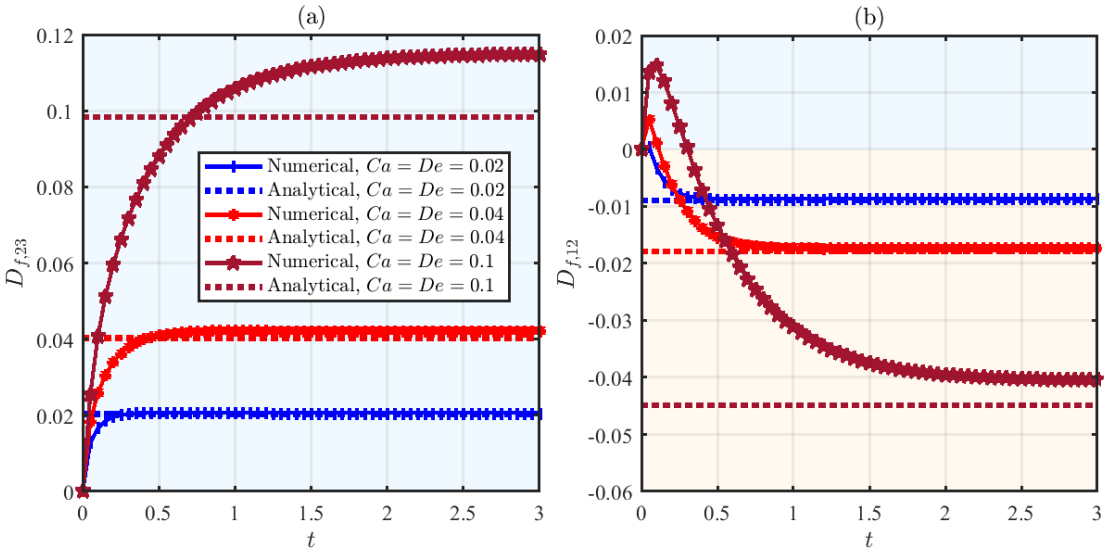

Figure S3: Temporal evolution of the deformation of the (a) shell and (b) core for three choices of Ca, De pairs, namely, $Ca = De = 0.02,\ 0.04$ and 0.1. The rest of the parameters are: $R = 0.5, \xi_1 = 2, \xi_3 = 0.5, \kappa = 1, \upsilon_{p,1} = \upsilon_{p,2} = 0.5, \upsilon_{p,3} = 0.1, \Lambda_1 = 1.5, \Lambda_3 = 0.1, \alpha_1 = \alpha_2 = \alpha_3 = 0.25$.

S3 Some Additional Results

S3.1 Transient deformation - numerical results

Figure S3 demonstrates the numerically obtained temporal evolution (markers) of the deformation factor:

$$D_{f,ij} = \frac{r_{ij,\text{axial}} - r_{ij,\text{lateral}}}{r_{ij,\text{axial}} + r_{ij,\text{lateral}}}$$

with $ij \equiv 23, 12$ (see Eq. (22) in the main article) for the shell (panel (a)) and the core (panel (b)) respectively, for three distinct combinations of Ca and De, namely, $Ca = De = 0.02,\ 0.04$ and 0.1. The regions of prolate (light blue) and oblate (light orange) shape deformations have been marked following the same color scheme as in the main article. The steady state asymptotic solutions for the deformation have also been shown in each panel for all three combinations of Ca and De as horizontal dotted lines. The values of other remaining parameters have been mentioned in the caption.

Panels (a) and (b) establish that the asymptotic solutions show reasonably good agreement with the numerical steady state deformations for both the shell and the core, when Ca and De are sufficiently small. For $Ca = 0.1$ (also $De = 0.1$) however, the asymptotic solutions underpredict the deformation, simply because at this Capillary number, the higher order corrections (i.e., $O(Ca^2)$ and others) also become important, which have been ignored in the present work. Even for Newtonian compound droplets, the leading order deformations are known [9] to be valid upto $Ca \approx 0.03 - 0.04$. It is perhaps intriguing to note from panel (b) that for $Ca \geq 0.04$, the core's deformation follows a non-monotonic trend in time. It initially becomes prolate ($D_{f,12} > 0$, light blue domain in the figures), followed by an increasingly oblate deformation before reaching steady state. The fact that non-monotonic trend persists even at low De values, suggests that this elastic stresses are not the primary reason for such variations in the deformation factor.

We further probe this start-up transience in Figure S4, where the deformation factors for the core and the shell are plotted with time at a fixed $Ca = 0.1$ and various choices of $De = 0.04,\ 0.1,\ 0.3$ and 1 - values of other relevant parameters have been mentioned in the caption. Panel (a) illustrates the deformation of the core and the shell till $t = 0.3$, while the variations in D_f till the steady state for the shell and the core have respectively been shown in panels (b) and (c). Two key observations may be made from panel (a). First, a relatively larger De results in a faster transient growth in D_f for the shell. However, at later times, the (see panel (b)), a crossover between the D_f curves corresponding to various De values are noted, which eventually results in a reduction in the deformation with increasing De - this is consistent with the discussion in the main article. Second, increasing De results in a relatively shorter lived prolate shape of the core (i.e., a faster onset of oblate deformation), although the precise value of De hardly has any impact on the extent of prolate deformation of the core when $t < 0.1$. Again, panel (c) reveals that at larger times, the core's steady state deformation eventually diminishes with De, in agreement with the discussion in §5.2 and §5.3 of the main article. The initial transient and the crossover between the deformation curves for various De values seem to be strongly dependent on the relaxation of the polymers in the three phases.

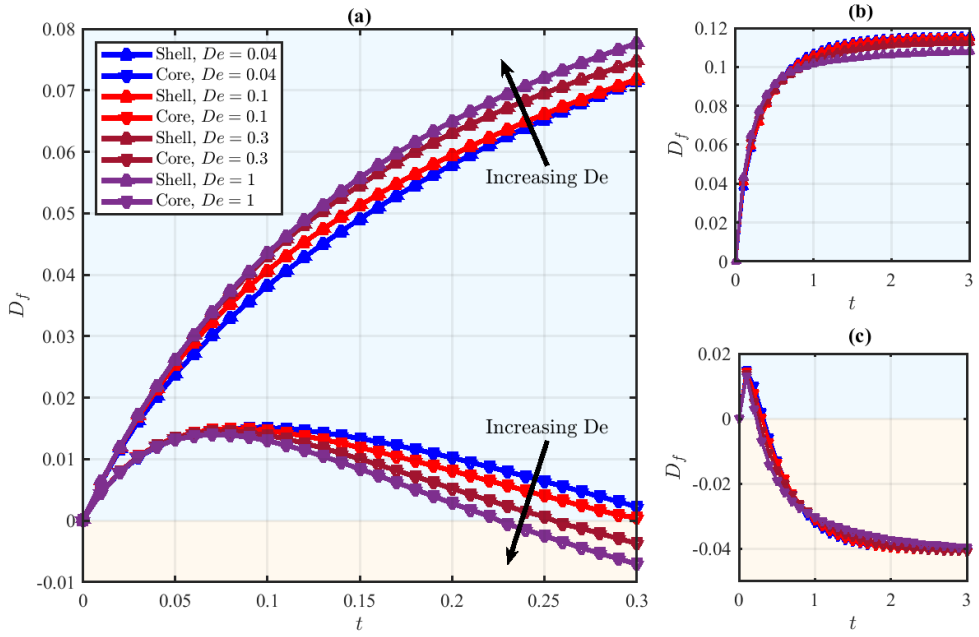

Figure S4: (a) Temporal evolution of the shell ($D_{f,23}$) and the core ($D_{f,12}$) from $t = 0-0.3$ for four choices of $De = 0.04, 0.1, 0.3, 1$. (b) Evolution of shell ($D_{f,23}$) and (c) core deformations ($D_{f,12}$) for the full simulation time $t = 0-3$. Other parameters are held constant at $Ca = 0.1, R = 0.5, \xi_1 = 2, \xi_3 = 0.5, \kappa = 1, \upsilon_{p,1} = \upsilon_{p,2} = 0.5, \upsilon_{p,3} = 0.1, \Lambda_1 = 1.5, \Lambda_3 = 0.1, \alpha_1 = \alpha_2 = \alpha_3 = 0.25$.

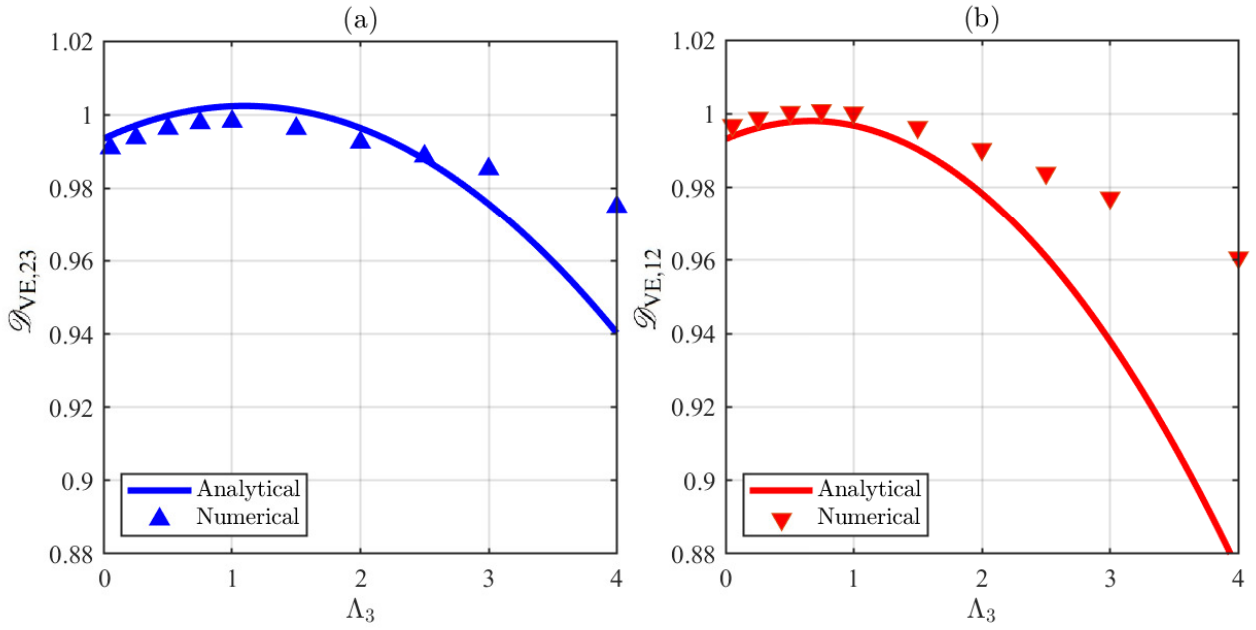

Figure S5: Variations in the normalized deformations of the (a) the shell and (b) the core at steady state against Λ_3. The asymptotic (line) and numerical (marker) solutions have been plotted for comparison. The normalized deformation is defined as: $\mathscr{D}_{\mathrm{VE}} = D_{f,ij}/D_{f,ij}^{\mathrm{Newt.}}$, where $D_{f,ij}^{\mathrm{Newt}}$ indicates the Newtonian deformation under nominally identical conditions. Other parameters are held constant at $Ca = 0.04, De = 0.1, R = 0.5, \xi_1 = 2, \xi_3 = 0.5, \kappa = 1, \upsilon_{p,1} = \upsilon_{p,2} = \upsilon_{p,3} = 0.5, \Lambda_1 = 2, \alpha_1 = \alpha_2 = \alpha_3 = 0.25$.

S3.2 Comparison between asymptotic and numerical results

Figure S5 shows an additional comparison between the asymptotic (line) and the numerical (markers) solutions for the steady state deformations, by plotting the normalized deformation factor $\mathscr{D}_{\mathrm{VE}} = D_{f,ij}/D_{f,ij}^{\mathrm{Newt.}}$ (shell - panel (a) and core - panel (b)) as a function of Λ_3 - the relaxation time of the suspending phase. It is evident that the two estimates show very good agreement with the largest error being less than 10% even when $\Lambda_3 = 4$. The agreement is better for the shell. More importantly, the non-monotonic variations in $\mathscr{D}_{\mathrm{VE}}$ with Λ_3 as depicted in Fig. 9 of the main article is also corroborated by the numerical simulations. Indeed, the maximum in $\mathscr{D}_{\mathrm{VE}}$ occurs at $\Lambda_3 \sim 1$ as per both asymptotic and numerical results.

S3.3 Consistency of the asymptotic results

We shall now try to compare the asymptotic solutions presented in §3.3 of the main article with prior studies and limiting cases from the literature. We first show that the leading order solutions (§3.3.1 in the main article) match exactly with those reported by Stone and Leal [9] for a Newtonian compound droplet in the small deformation limit. For comparison, we note that they had denoted the matrix fluid as phase-1, the shell fluid as phase-2, and the core fluid as phase-3, whereas in our work the order is reversed. Therefore, to avoid confusion, we use the subscripts χ_c, χ_s, and χ_m to respectively denote the core, the shell, and the matrix properties, while the interfaces between core-shell and shell-matrix fluids are respectively denoted using the subscripts *cs* and *sm*. We use a superscript χ^P to denote parameters as defined in the present manuscript, distinguishing them from the ones defined in previous works, which are kept unchanged for ease of correlation. Thus, the viscosity ratios $\lambda_{21} = \mu_s'/\mu_m'$ and $\lambda_{32} = \mu_c'/\mu_s'$ relate to the current definitions as $\lambda_{21} = 1/\xi_3^P$ and $\lambda_{32} = \xi_1^P$. We further define $\Omega = \sigma_{cs}'/\sigma_{sm}' = \kappa^P$ as the ratio of surface tensions at the two interfaces. The radius ratio K (R in the present work, termed K in this document to avoid confusion with the ratio of surface tension) has the same definition as $K = R^P = \frac{R_c'}{R_s'}$. The outer capillary number was defined by Stone and Leal as $C_o = \frac{G' R_s' \mu_m'}{\sigma_{sm}'}$ which is slightly different from the present definition of Ca^P; the two are related as $C_o = \lambda_{21} Ca^P$. Finally, Stone and Leal defined the interfaces as:

$$r_{sm} = 1 + A_{21} \frac{\mathbf{x}\cdot\mathbf{E}\cdot\mathbf{x}}{r^2} \tag{S8a}$$

$$r_{cs} = K\left[1 + A_{32} \frac{\mathbf{x}\cdot\mathbf{E}\cdot\mathbf{x}}{r^2}\right], \tag{S8b}$$

where $\frac{\mathbf{x}\cdot\mathbf{E}\cdot\mathbf{x}}{r^2}$ simplifies to $P_2(\eta)$ in the spherical coordinate system with $\eta = \cos(\theta)$ used in our work. Comparing the definition of the interface with Eq. (6) of the main article, and considering only the $O(Ca)$ deformations, it is clear that the following equations must be identical to Eq. (S8)

$$r_{sm}^P = 1 + Ca^P \, \mathscr{S}_{cs,2}^{(0,1)P} \, P_2(\eta) \tag{S9a}$$

$$r_{cs}^P = R^P + Ca^P \, \mathscr{S}_{sm,2}^{(0,1)P} \, P_2(\eta) \tag{S9b}$$

Thus, it follows directly that

$$\mathscr{S}_{cs,2}^{(0,1)P} = \frac{K}{\lambda_{21} C_o} A_{32} \tag{S10a}$$

$$\mathscr{S}_{sm,2}^{(0,1)P} = \frac{1}{\lambda_{21} C_o} A_{21} \tag{S10b}$$

It may be verified that the equality holds true by using Maple files provided as part of this Supplementary Material. One may refer to §S4 to understand how to employ the Maple files to arrive at explicit solutions. Below, we show some of the limiting cases for comparison:

1. Vanishingly Small Core: $R^P = K \to 0$

   $$\lim_{K\to 0} A_{21} = \frac{1}{8}\frac{16 + 19\lambda_{21}}{\lambda_{21} + 1}$$

   which corresponds to the well known first order deformation of a simple droplet in extensional flow [10]. The present asymptotic solution at leading order yields:

   $$\lim_{R^P\to 0} \mathscr{S}_{sm,2}^{(0,1)P} = \frac{1}{8}\frac{\xi_3^P(16\xi_3^P + 19)}{\xi_3^P + 1}$$

   which satisfies Eq. (S10) and also matches exactly with the above result because $\xi_3^P = 1/\lambda_{21}$. The limit of $R^P = K \to 0$ may be applied to higher order solutions presented in §3.3.2 and §3.3.3 of the main article as well to mimic the case of a simple viscoelastic droplet placed in an extensional flow.

2. Unit Viscosity Ratios: when $\lambda_{21} = \lambda_{32} = 1$, Stone and Leal's result for A_{32} reduces to (see Eq. (S8)):

   $$A_{32,\lambda_i=1} = -\frac{105}{4}\frac{C_o(K+1)K}{\Omega(9K^6 + 9K^5 - 33K^4 - 33K^3 + 16K^2 + 16K + 16)}$$

   In our work, this same limit entails $\xi_1^P = \xi_3^P = 1$, and leads to (in Eq. (S10)):

   $$\mathscr{S}_{cs,2}^{(0,1)P} = -\frac{105}{4}\frac{R^{P2}(R^P+1)}{\kappa^P(9R^{P5} + 18R^{P4} - 15R^{P3} - 48R^{P2} - 32R^P - 16)(R^P - 1)}$$

   It is straightforward to verify with the help of Eq. (S10) that the above two results are identical.

S4 Analytical expressions through Maple™ files

This section presents a step-wise guide on how to use the Maple™[11] ('.mw') files appended with this document within the Supplementary Material, to arrive at the expressions for the various constants and coefficients appearing in §3.3 of the main article. The following Maple™ files are available as part of the supplementary material:

1. "Order 1 solution.mw": calculates the leading order flow field
2. "Order Ca deformation.mw": calculates the leading order deformations
3. "Order De flowfield.mw": calculates the $O(De,0)$ flow field
4. "Order DeCa deformation.mw": calculates the $O(De,Ca)$ deformations
5. "Order De2 flowfield.mw": calculates the $O(De^2,0)$ flow field
6. "Order De2Ca deformation.mw": calculates the $O(De^2,Ca)$ deformations

All files are required to be saved within the same directory/folder. To reproduce the results and arrive at the explicit expressions of the various flow-parameters, the files need to be only slightly modified depending on the desired results and run in the a sequence as enumerated below. Each file also contains additional comments for better understanding.

1. Run "Order 1 solution.mw"
   (a) The file begins with a short instructional header, after which the parameters are to be specified. Keeping a parameter definition commented will preserve that parameter as a variable included in the final expressions (refer to the in-file instructions). Note that keeping all the parameters undefined (i.e., without assigning numerical value) is possible, but the expressions thus obtained will be extremely complicated, especially for the higher order solutions which cannot be coherently displayed in the Maple environment.
   (b) Running this file produces two output files in '.txt' format. i) "parameters.txt" which saves the parameters, if specified, and will be read by the subsequent files to reduce repetition. ii) "LeadingOrderConstants.txt" which saves the solved expressions of the constants appearing in the stream functions pertaining to the parameters specified.
   (c) To evaluate any flow variable such as the velocity or the stress components, the commands after the leading order constants are solved for, at the end of the file, are to be used. Details for doing the same are included in the file.
2. Run "Order Ca deformation.mw"
   (a) This file uses the output files of the previous step and implements the normal stress boundary condition to calculate the $O(0,Ca)$ deformations.
   (b) If specific parameters are not defined in the previous step, i.e., if all the parameters are considered to be variables to appear in the explicit expressions, comment out the reading of the "parameters.txt" as instructed in the file.
   (c) This file produces one output file named "LeadingOrderSfunctions.txt" which includes the values/expressions calculated for the deformation functions.
3. Run "Order De flowfield.mw"
   (a) This file uses the output files "parameters.txt", "LeadingOrderConstants.txt" generated using previous steps, and calculates the $O(De,Ca)$ stream functions for the three phases. Non-zero modes are highlighted in yellow in the file for ease of reference.
   (b) The file follows a similar format to the "Order 1 solution.mw" with an added portion dedicated to solving the particular solutions to the stream function equation as it becomes non-homogeneous at $O(De)$.
   (c) Once the particular solutions to stream functions of each phase is calculated, the file follows the same format as in step-1 to calculate the constants appearing in the general solution of the stream functions.
   (d) This file produces one output file termed "OrderDeConstants.txt" which saves the values/expressions of the constants appearing in the stream function solution (both, the particular constants - characterized by double letters i.e. *AA_De*1_1 and similar, and the ones in the general soln. with single letters i.e, *A_De*1_1 and similar.
4. Run "Order DeCa deformation.mw"
   (a) Follows the same format as in step-2. Non-zero modes are highlighted in yellow.

(b) Utilizes the '.txt' files generated from steps 1,3

(c) Produces the output file "OrderDeCaSfunctions.txt" containing the deformation functions at order $O(De, Ca)$

5. Run "Order De2 flowfield.mw"

   (a) Uses the files "parameters.txt", "LeadingOrderConstants.txt", "OrderDeConstants.txt" to calcualte the constants appearing in the stream functions at order $O(De^2, 0)$.

   (b) Follows a format similar to the ones described in step-3.

   (c) Saves the values/expressions of the constants appearing in the stream function solution in a file named "OrderDe2Constants.txt"

6. Run "Order De2Ca deformation.mw"

   (a) Follows the same format as in step-4.

   (b) Calculates deformation at order $O(De^2, Ca)$ using the '.txt' files generated from steps 1,3,5.

   (c) Saves the deformation functions in an output file named "OrderDe2CaSfunctions.txt"

References

[1] Comsol multiphysics® v. 5.6, stockholm, sweden, . URL `http://www.comsol.com`.

[2] Dilip Rajagopalan, Robert C. Armstrong, and Robert A. Brown. Finite element methdos for calculation of steady, viscoelastic flow using constitutive equations with a newtonian viscosity. *Journal of Non-Newtonian Fluid Mechanics*, 36:159–192, 1990. ISSN 0377-0257. doi: https://doi.org/10.1016/0377-0257(90)85008-M. URL `https://www.sciencedirect.com/science/article/pii/037702579085008M`.

[3] Franck Boyer, Céline Lapuerta, Sebastian Minjeaud, Bruno Piar, and Michel Quintard. Cahn–hilliard/navier–stokes model for the simulation of three-phase flows. *Transport in Porous Media*, 82(3):463–483, 2010.

[4] Franck Boyer and Céline Lapuerta. Study of a three component cahn-hilliard flow model. *ESAIM: Mathematical Modelling and Numerical Analysis*, 40(4):653–687, 2006.

[5] Cfd module user's guide, pp. 501-504, comsol multiphysics® v. 5.6, stockholm, sweden, . URL `http://www.comsol.com`.

[6] Vittorio E Badalassi, Hector D Ceniceros, and Sanjoy Banerjee. Computation of multiphase systems with phase field models. *Journal of computational physics*, 190(2):371–397, 2003.

[7] Hanswalter Giesekus. A simple constitutive equation for polymer fluids based on the concept of deformation-dependent tensorial mobility. *Journal of Non-Newtonian Fluid Mechanics*, 11(1-2):69–109, 1982.

[8] David Jacqmin. Calculation of two-phase navier–stokes flows using phase-field modeling. *Journal of computational physics*, 155(1):96–127, 1999.

[9] HA Stone and LG Leal. Breakup of concentric double emulsion droplets in linear flows. *Journal of Fluid Mechanics*, 211:123–156, 1990.

[10] L Gary Leal. *Cambridge series in chemical engineering: Advanced transport phenomena: Fluid mechanics and convective transport processes series number 7*. Cambridge University Press, Cambridge, England, June 2012.

[11] Maplesoft, a division of Waterloo Maple Inc.. Maple. URL `https://www.maplesoft.com/`.