

**Direct Demonstration of Self-Similarity In a Hydrodynamic
Treatment of Polymer Self-Diffusion**

by

Susan Carol Merriam

A Thesis

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Master of Science

in

Physics

by

May 2002

APPROVED:

Professor George D. J. Phillis, Thesis Advisor

Professor Thomas H. Keil, Head of Department

Contents

1	Introduction	1
2	Kirkwood-Riseman Model	4
3	Phillies Model	9
3.1	Multichain Hydrodynamic Interaction Tensors	9
3.2	Ensemble Average of the Hydrodynamic Interaction Tensors	24
4	Derivation of $\mathbf{u}^{(4)}$	27
4.1	Analysis	27
4.2	Numerical Confirmation of $\mathbf{u}^{(4)}$	32
5	Evaluation of \mathbf{T}_{54321} and \mathbf{b}_{1232}	33
6	Ensemble Average of the Self-Interaction Tensor for the Figure-Eight Scattering Diagram	35
6.1	Analysis	35
6.2	Monte Carlo Routine	37

7 Discussion	39
A polyalg.for	43
B hydrpoly.for	50
C flyind.for	64
D flyintegrat.com	82

Abstract

The self-diffusion coefficient of a polymer in solution may be expanded in the concentration of the polymer, as seen in equation 1. The linear term would represent a perturbation due to the presence of another polymer; the c^2 term would represent a perturbation due to interactions of trios of polymers. Phillies[1] determined the c^2 term of a virial expansion of the self-diffusion coefficient for trios of polymers interacting via a ring. Here I determine a correction to the c^2 term due to trios of polymers interacting via a figure-eight scattering diagram: the equivalent of four polymers interacting in a ring where the second polymer and the fourth polymer are the same.

$$D_s(c) = D_0(1 + \alpha D_0 c + \beta D_0^2 c^2 + \dots), \quad (1)$$

or,

$$D_s(c) = D_0(1 + \alpha D_s(c) c). \quad (2)$$

A D_0 may be replaced by $D_s(c)$ in equation 1 to arrive at equation 2. The left-hand-side of equation 2 is the final self-diffusion coefficient, and the $D_s(c)$ on the right-hand-side of this equation is that due to the question of self-similarity. If the $D_s(c)$ on the right-hand-side is given by equation 1, resulting in $\beta = \alpha^2$, it may be said that the system exhibits self-similarity. I demonstrate self-similarity

quantitatively for a polymer solution using a generalized Kirkwood-Riseman model of polymer dynamics.

The major physical assumption of the model I utilize to derive equation 2 is that, in solution, polymer motions are dominantly governed by hydrodynamic interactions between the chains. First, I review the Kirkwood-Riseman model for intrachain hydrodynamic interactions. I then discuss Phillies' extension of this model to interchain interactions for duos or trios of polymers in a ring. I analytically calculate the hydrodynamic interaction tensor from a multiple scattering picture \mathbf{T}_{54321} , for five polymers in solution and verify this tensor by numerical differentiation. Finally, I perform the ensemble average of the self-interaction tensor \mathbf{b}_{1232} appropriate to the figure-eight scattering diagram both analytically and with a Monte Carlo routine, thereby verifying equation 2 to second order in concentration.

Table of Symbols

A	scalar part of $\mathbf{u}^{(3)}$ less $\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)}$
a	$R_{h1}R_{h2}R_{h3}R_{g2}^2R_{g3}^2/R_1^2R_2^3$
a_0	bead-bead distance of closest approach
B_2, B_3	leading virial coefficients
B'_3	virial coefficient due to figure-eight scattering diagram
b	$\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2$
\mathbf{b}_l	bond vector
b_0	length of bond vector
$\mathbf{b}_{il\dots}$	self-interaction tensor
c	concentration or $\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3$
D_0	zeroth order self-diffusion constant
$\dot{\mathbf{D}}_j$	velocity of bead j due to changes in polymer internal configuration
D_s	self -diffusion coefficient
d	$\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_3$
\mathbf{F}_l	force applied to bead l
f_j, f_{0j}	bead drag coefficient
$g(s)$	bead distribution function
$g^{(n)}$	n -polymer distribution function

k_B	Boltzmann constant
M	molecular weight of polymer or number of beads in a chain
M_0	molecular weight of monomer
N	number of chains
R_g	radius of gyration
R_h	hydrodynamic radius
\mathbf{s}	position in polymer relative to its center-of-mass
T	temperature
\mathbf{T}_{ls}	Oseen tensor
$\mathbf{T}_{i\dots j}$	cross-interaction tensor
$\mathbf{u}^{(n)}$	current induced by n th polymer
\mathbf{u}_l	velocity of monomer l
$\mathbf{V}^{(n)}$	translational velocity of polymer n
\mathbf{v}_0	velocity field of solvent
\mathbf{v}'	disturbance in solvent
\mathbf{v}_j	velocity of bead j relative to stationary solvent
\mathbf{w}_l	$\mathbf{v}_l - \mathbf{u}_l$

α	$\alpha D_0 = B_2$
β	$\beta D_0^2 = B_3$
γ	fixed skeletal angle
η	viscosity
μ	mobility
Ξ	molecular friction constant
ξ	friction constant
ϕ_l	angle between planes of successive bond pairs
$\boldsymbol{\Omega}^{(n)}$	angular velocity of chain n
$\nabla_{\mathbf{R}}$	gradient with respect to the highest numbered of the variables of the function being acted upon.

Chapter 1

Introduction

This thesis considers hydrodynamic interactions between polymers in solution and how they determine the self-diffusion coefficient of a polymer. Originally, Kirkwood and Riseman[2] studied a single polymer, and how its molecular structure would relate to its diffusion constant. Phillies[1] extended Kirkwood and Riseman's theory to interactions between polymers. Here, interactions between quintets of polymers in solution, using Phillies' model, are studied. The coefficient in a virial expansion of the self-diffusion coefficient due to the addition of a figure-eight scattering diagram of trios of polymers is determined.

In the Kirkwood-Riseman model[2], the diffusion constant of a polymer is related to its molecular structure. This model considers hydrodynamic shielding of inner monomers of high molecular weight polymers in the random-coil model. Flow in the solvent field is perturbed by the introduction of the polymer. This disturbance is described by the Oseen tensor. Forces due to drag are experienced by the monomers.

An ensemble average of these forces is equal to the product of a molecular friction constant and the velocity of the center-of-mass of the polymer. Using the Einstein relation, the diffusion constant is subsequently determined.

Phillies' model[1], an extension of the Kirkwood-Riseman use of the Oseen tensor to multiple polymers, uses a generalized Einstein relation with μ , a mobility, due to single polymers, and to duos and trios of polymers in rings. The mobility is expressed as a sum of self-interaction tensors. These tensors are determined from cross-interaction tensors by requiring the first and last polymers to be the same. The cross-interaction tensors are determined from the scattering wake of a series of polymers interacting hydrodynamically. The theoretical basis of the wake is the Oseen tensor.

In Phillies' model, three forces are considered, namely, the hydrodynamic coupling force, the excluded volume, and the bond-coupling force. In this model the Oseen tensor interaction is expanded in powers of the hydrodynamic radius and subsequently subject to zero-net-force and zero-net-torque conditions to determine the translational and rotational velocities of the polymer. This polymer then scatters the applied hydrodynamic flow. Phillies' derivation produces general relations for the translational and rotational velocities of the $(n + 1)$ st polymer in terms of the scattered wake of the n th polymer. From the generalized Einstein relation which involves an ensemble average over the mobility, Phillies determines the leading coefficients in a virial expansion of the self-diffusion coefficient due to rings of duos or trios of polymers interacting hydrodynamically.

In the following, $\boldsymbol{\Omega}^{(4)}$, the angular velocity of a fourth polymer due to the wake scattered by the third polymer in a scattering series is derived. Then $\mathbf{u}^{(4)}$, the flow field scattered by the fourth polymer, is determined analytically and numerically. It is here shown that Phillies' expression for $\mathbf{u}^{(n+1)}$ can be simplified. \mathbf{T}_{54321} , the cross-interaction tensor for quintets of polymers interacting hydrodynamically, is calculated. By requiring the first and fifth polymers, and the second and fourth polymers to be the same, the self-interaction tensor, \mathbf{b}_{1232} , for a multiple scattering picture for the figure-eight scattering diagram was derived from \mathbf{T}_{54321} . The expression was integrated analytically and with a Monte Carlo routine to determine its ensemble average. This ensemble average is the primary part of the c^2 coefficient in the virial expansion of the self-diffusion coefficient.

Chapter 2

Kirkwood-Riseman Model

The Kirkwood-Riseman model[2] relates intrinsic viscosities and diffusion constants of flexible macromolecules to molecular structure. Prior to this model, Staudinger's Law[3] predicted that intrinsic viscosities are linearly proportional to the molecular weight M . While Kuhn[4], Huggins[5] and Kramers[6] confirmed this Law theoretically for polymers of random coil structure, experiment found that viscosity goes as M^α , $0.5 \leq \alpha \leq 1.0$. Debye[7] suggested that the reason for the discrepancy between theory and experiment may have been due to assuming equal resistance due to each monomer unit, ignoring interactions between monomers within the chain. Debye proposed that, at high molecular weight, hydrodynamic shielding of the interior monomers may become so effective that they contribute little resistance to the resistance offered by the entire molecule to an external fluid flow. In the Kuhn[4], Huggins[5] and Kramers[6] models, each element was taken to offer the same resistance to solvent flow that it would have offered if there were no other chain elements

in its neighborhood. To describe the reduced flow due to shielding, Debye[8] and Brinkman[9] independently investigated a molecular model of a sphere with uniformly distributed resisting points equal in number to the number of monomers. Debye's model agrees quantitatively with the experimental dependence of the intrinsic viscosity on molecular weight.

In the Kirkwood-Riseman model[2], the reduced flow is based on a more realistic random-coil model. Below the method and model used by Kirkwood and Riseman to determine intrinsic viscosities and translation diffusion constants of flexible macromolecules is presented. The results agree qualitatively with those of Debye. However, the model is thought[2] to be quantitatively more reliable since the molecular model employed is more accurate than that of Debye.

The Kirkwood-Riseman model uses the random coil model to treat fluid flow through a linear polymer $(\text{CHX})_{2n+1}$. In the notation for the random coil model, skeletal carbon atoms are numbered $-n$ to n , and are connected by $2n$ bond vectors \mathbf{b}_l from skeletal atom $l - 1$ to atom l . The length of these bond vectors $|\mathbf{b}_l|$ equals b_0 . The internal configuration of the macromolecule is specified by ϕ_l , the angles between the planes of successive bond pairs, and γ , the fixed skeletal bond angle. The configuration changes with internal rotation about each bond. This rotation is hindered, with a plane of symmetry at $\phi_l = 0$. The degree of polymerization is $2n$, represented by $z = M/M_0$, where M is the total molecular weight, and M_0 is the molecular weight of the monomer unit CHX.

In the random coil model, average values for \mathbf{R}_{ls}^2 , \mathbf{s}_l^2 and $\mathbf{s}_l \cdot \mathbf{s}_s$ are obtained,

where R_{ls} is the distance between chain elements l and s , and \mathbf{s}_l is the position of chain element l with respect to the molecular center-of-mass. In addition, for a normally distributed R_{ls} the average value for $1/R_{ls}$ as a function of mean bond length and l and s is obtained.

When a polymer is introduced to a low molecular weight fluid in which a velocity field $\mathbf{v}_0(\mathbf{R})$ exists, the flow is perturbed by the resistance of each monomer unit. A force \mathbf{F}_l is applied to the solvent by monomer l as follows: if \mathbf{v}_l is the velocity the solvent would possess at the location of monomer l without monomer l , and \mathbf{u}_l is the velocity of that monomer, \mathbf{F}_l is given by

$$\mathbf{F}_l = -\xi_l \mathbf{w}_l, \quad \mathbf{w}_l = \mathbf{v}_l - \mathbf{u}_l. \quad (2.1)$$

Here ξ_l is the friction constant particular to the monomer structure and the solvent.

The disturbance \mathbf{v}' in the solvent at \mathbf{R}_l , the location relative to the monomer l at which \mathbf{F}_l is applied, is determined via the Oseen tensor \mathbf{T}

$$\mathbf{v}'(\mathbf{R}_l) = \mathbf{T}(\mathbf{R}_l) \cdot \mathbf{F}_l \quad (2.2)$$

$$\mathbf{T}(\mathbf{R}_l) = 1/(8\pi\eta_o R_l)[\mathbf{I} + \hat{\mathbf{R}}_l \hat{\mathbf{R}}_l] \quad (2.3)$$

Here η_o is the viscosity of the solvent, \mathbf{I} is the unit tensor, and $\hat{\mathbf{R}}_l$ is the unit vector in the direction of \mathbf{R}_l .

The tensor \mathbf{T} appears twice in the Kirkwood-Riseman model. First, it is em-

ployed to determine the disturbance in the flow at monomer l due to other portions of the molecule. Second, it is used to calculate an overall disturbance in the solvent at a specified distance from the molecular center-of-mass due to all monomers. The velocity \mathbf{v}_l the solvent would have had without monomer l is given by

$$\mathbf{v}_l = \mathbf{v}_l^o - \xi \sum_{\substack{s=-n \\ s \neq l}}^n \mathbf{T}_{ls} \cdot \mathbf{w}_s. \quad (2.4)$$

Here \mathbf{v}_l^o is the solvent velocity without a polymer and \mathbf{T}_{ls} is the Oseen tensor corresponding to R_{ls} .

If \mathbf{u}_l is given,

$$\mathbf{F}_l = -\xi(\mathbf{v}_l^o - \mathbf{u}_l) - \xi \sum_{\substack{s=-n \\ s \neq l}}^n \mathbf{T}_{ls} \cdot \mathbf{F}_s. \quad (2.5)$$

The perturbation in the solvent flow field \mathbf{v}' produced by the molecule as a whole at a position \mathbf{R} relative to the molecular center-of-mass is given by

$$\mathbf{v}'(\mathbf{R}) = \frac{1}{8\pi\eta_0} \sum_{l=-n}^n \left\{ \frac{\mathbf{F}_l}{|\mathbf{R} - \mathbf{s}_l|} + \frac{(\mathbf{R} - \mathbf{s}_l)(\mathbf{R} - \mathbf{s}_l) \cdot \mathbf{F}_l}{|\mathbf{R} - \mathbf{s}_l|^3} \right\}. \quad (2.6)$$

For the translational diffusion constant, the center-of-mass of the polymer is considered to have a velocity \mathbf{u} relative to the solvent. The molecular friction constant Ξ is given by

$$\langle \mathbf{F} \rangle = \Xi \mathbf{u}, \quad \langle \mathbf{F} \rangle = - \sum_{l=-n}^n \langle \mathbf{F}_l \rangle. \quad (2.7)$$

The average force $\langle \mathbf{F} \rangle$ on the molecular center-of-mass due to the solvent is the

sum of the average forces – $\langle \mathbf{F}_l \rangle$ exerted by the individual monomers. Once the friction constant is known, the Einstein relation

$$D = kT/\Xi \tag{2.8}$$

determines the diffusion constant.

Chapter 3

Phillies Model

3.1 Multichain Hydrodynamic Interaction Tensors

In Phillies' generalization[1] of the Kirkwood-Riseman model[2] hydrodynamic interactions between polymer chains are considered. The Kirkwood-Riseman model of a bead-bead Oseen interaction is extended in the Phillies model to include hydrodynamic interactions from a multiple scattering picture, between two, three or four polymer chains. The chain-chain interaction is an average over a sum of bead-bead interchain interactions. In Phillies' model three forces are utilized: the hydrodynamic coupling of nearby beads, the excluded volume, and the bond-coupling force linking neighboring beads on a polymer chain.

The first term of an expansion of the interaction is the Oseen tensor, an application proposed in the Kirkwood-Riseman model for interactions between monomers in one polymer. Since the Oseen tensor represents a long-range force ($\propto 1/R$), where

R is the distance from the center-of-mass, other shorter-range forces such as in the Rotne-Prager model have been proposed. Shorter-range forces would be given by higher-order expansions in s/R , where s is the distance of a bead from the molecular center-of-mass.

In Phillies' model[1], hydrodynamic interactions between distinct chains are given by self- and cross-interaction tensors, $\mathbf{b}_{jl\dots}$ and $\mathbf{T}_{i\dots j}$, respectively. Instead of the specific polymer/monomer designation of the Kirkwood-Riseman model[2], Phillies considers beads or "frictional point", as opposed to monomers, as the salient unit of a polymer.

Phillies restricts his model to interchain coupling. Intrachain coupling is taken to act so as to entrain the solvent. Entrainment causes f_{0i} , the true drag coefficient of the entire chain, to be much less than would be expected from summed individual bead contributions. Phillies considers explicitly only scattering events between beads on distinct chains. A distance of closest approach is imposed by a radial distribution function. Bond interactions, which determine the stiffness of a chain, are incorporated into the radius of gyration and the hydrodynamic radius of each chain. The further physical assumption is that each chain's motion is highly overdamped. The inertia and moment of inertia of a chain are negligible. For a given chain, the sum of forces produced by the constituent beads is zero; the sum of torques about the center-of-mass of a given polymer produced by the constituent beads is also zero.

Hydrodynamic interactions arise because each bead has a frictional interaction

with the solvent. A bead j imparts a force on the solvent according to

$$\mathbf{F}_j = f_j(\mathbf{v}_j - \mathbf{u}(\mathbf{r}_j)). \quad (3.1)$$

Here $\mathbf{u}(\mathbf{r}_j)$ is the velocity that the solvent would have had without presence of bead j , $\mathbf{v}_j = \partial \mathbf{r}_j / \partial t$ is the velocity of bead j , and f_j is the bead drag coefficient, where $f_j = 6\pi\eta R_h$, η being the solvent viscosity and R_h is the bead's effective hydrodynamic radius. The resulting induced current in the solvent, which leads to hydrodynamic coupling of pairs of beads, is given to lowest order in R_h by

$$\mathbf{u}^{(i)}(\mathbf{r}) = \mathbf{T} \cdot \mathbf{F}_j, \quad (3.2)$$

$$\mathbf{u}^{(i)}(\mathbf{r}) = \frac{3R_h}{4r}(\mathbf{I} + \hat{\mathbf{r}}\hat{\mathbf{r}}) \cdot \mathbf{v}_j. \quad (3.3)$$

Here \mathbf{T} is the Oseen tensor. This application of the tensor was proposed by Kirkwood and Riseman[2]. The superscript i refers to chain i , subsequent in the scattering series to the chain which has bead j .

Hydrodynamic interactions between entire chains are described by the self- and cross-interaction tensors \mathbf{b}_{jl} and \mathbf{T}_{ij} and their multi-chain analogues. For instance, \mathbf{b}_{jlm} would account for the self-interactions in the diffusion coefficient of chain j due to the presence of chains l and m . Likewise, \mathbf{T}_{ilj} would account for the cross-interactions in the diffusion coefficient of chains i and j due to the presence of chain l . If an external force \mathbf{F}_j is applied to the beads of a freely floating chain j giving

this chain a uniform translational velocity $\mathbf{V}^{(1)}$, each of the other chains i , $i \neq j$, acquires an induced velocity from chain j 's wake determined by \mathbf{T}

$$\mathbf{v}_i = [\mathbf{T}_{ij} + \sum_{\substack{l=1 \\ l \neq i,j}}^N \mathbf{T}_{ilj} + \dots] \cdot \mathbf{F}_j. \quad (3.4)$$

Here \mathbf{F}_j is the total force acting on chain j . Furthermore, the induced velocity of chain j is affected by the presence of other chains through \mathbf{b}

$$\mathbf{v}_j = f_{0j} [\mathbf{I} + \sum_{\substack{l=1 \\ l \neq j}}^N \mathbf{b}_{jl} + \sum_{\substack{l,m=1 \\ l,m \neq j}}^N \mathbf{b}_{jlm} + \dots] \cdot \mathbf{F}_j. \quad (3.5)$$

Phillies' treatment of polymer diffusion neglects intrachain hydrodynamic interactions. As a consequence, the computed chain-chain and chain-chain-chain hydrodynamic interactions overestimate the actual interactions.

The bead-bead excluded volume force prevents pairs of beads from overlapping; the centers of two beads may never be closer than $2a_0$. If beads could interpenetrate, no solvent could flow in the space along the line of centers. This situation is not taken into account in the derivation of the Oseen and other bead-bead hydrodynamic interaction tensors. In addition, without a distance of closest approach, many integrals of the derivation presented here would be divergent, a non-physical result.

The calculation of the interaction tensors \mathbf{b} and \mathbf{T} begins with a polymer i which has a translational velocity $\mathbf{V}^{(i)}$, a rotational velocity $\boldsymbol{\Omega}^{(i)}$ with respect to the

stationary solvent, and $\dot{\mathbf{D}}_j$, the part of bead j 's velocity due to the time dependence of the polymer's internal coordinates. Bead j of chain i has an initial velocity

$$\mathbf{v}_j = \mathbf{V}^{(i)} + \boldsymbol{\Omega}^{(i)} \times \mathbf{s}_j + \dot{\mathbf{D}}_j \quad (3.6)$$

with respect to the solvent. This bead exerts a force on the solvent $\mathbf{F}_j = f_j \mathbf{v}_j$. Utilizing the Oseen tensor, the induced current in the solvent $\mathbf{u}^{(1)}(\mathbf{r})$ due to the entire chain is

$$\mathbf{u}^{(1)}(\mathbf{r}) = \sum_{i=1}^M \mathbf{T}(\mathbf{r} - \mathbf{s}_i) \cdot f_i \mathbf{v}_i^{(1)}. \quad (3.7)$$

Here M is the number of beads in a polymer.

Next the set of bead positions \mathbf{s}_i is replaced by a variable \mathbf{s} , the position relative to the center-of-mass, and a bead distribution function $g(s)$, which is centered on the center-of-mass of the polymer in question. This distribution function gives an ensemble average density of beads at a distance s from the polymer center-of-mass. Here $\int ds g(s) s^2 = R_g^2$, R_g being the radius of gyration of the polymer. There is evidence that the configuration of a typical polymer is other than spherically symmetric[1]. An ensemble average of the configuration nonetheless leads to $g(s)$ as a function of $s = |\mathbf{s}|$ only. For a higher order expansion of \mathbf{T} in s/r , greater caution in the treatment of $g(s)$ would be necessary. In addition to the introduction of s , the sum of the bead drag coefficients f_i is replaced by a chain drag coefficient f_0 .

A MacLaurin series expansion for the Oseen tensor is $\mathbf{T}(\mathbf{r} - \mathbf{s}_i) = \mathbf{T}(\mathbf{r}) - (\mathbf{s}_i \cdot$

$\nabla)\mathbf{T}(\mathbf{r}) + O(s^2)$, or

$$\mathbf{T}(\mathbf{r} - \mathbf{s}) = \frac{1}{8\pi\eta} \left[\frac{\mathbf{I} + \hat{\mathbf{r}}\hat{\mathbf{r}}}{r} - \hat{\mathbf{r}} \frac{\mathbf{s} \cdot (\mathbf{I} - 3\hat{\mathbf{r}}\hat{\mathbf{r}})}{r^2} - \frac{\mathbf{s}\hat{\mathbf{r}}}{r^2} + \frac{\mathbf{s} \cdot \hat{\mathbf{r}}}{r^2} \mathbf{I} \right] + O\left(\left(\frac{s}{r}\right)^2\right). \quad (3.8)$$

Here $\mathbf{s}\hat{\mathbf{r}}$ is the dyadic product of \mathbf{s} and $\hat{\mathbf{r}}$. Placing equation 3.8 into equation 3.7, and using the replacements in the previous paragraph, it is found that

$$\mathbf{u}^{(1)} = f_0 \int d\mathbf{s} g(s) \left\{ \frac{1}{8\pi\eta} \left[\frac{\mathbf{I} + \hat{\mathbf{r}}\hat{\mathbf{r}}}{r} - \hat{\mathbf{r}} \frac{\mathbf{s} \cdot (\mathbf{I} - 3\hat{\mathbf{r}}\hat{\mathbf{r}})}{r^2} - \frac{\mathbf{s}\hat{\mathbf{r}}}{r^2} + \frac{\mathbf{s} \cdot \hat{\mathbf{r}}}{r^2} \mathbf{I} \right] \right\} \cdot \{ \mathbf{V}^{(1)} + \boldsymbol{\Omega}^{(1)} \times \mathbf{s}^{(1)} \}, \quad (3.9)$$

neglecting internal mode relaxation.

For the longest range parts of the solvent flow (zeroth and first orders in the s/r expansion of \mathbf{T}), using $\int d\mathbf{s} g(s) \mathbf{s} \cdot \hat{\mathbf{r}} \boldsymbol{\Omega} \times \mathbf{s} = (R_g^2/3) \boldsymbol{\Omega} \times \hat{\mathbf{r}}$, that $f_0 = 6\pi\eta R_{h1}$, and that terms odd in \mathbf{s} vanish with integration over \mathbf{s}

$$\mathbf{u}^{(1)}(\mathbf{r}) = \frac{3 R_{h1}}{4 r} [\mathbf{I} + \hat{\mathbf{r}}\hat{\mathbf{r}}] \cdot \mathbf{V}^{(1)} + \frac{1}{2} \frac{R_{h1} R_{g1}^2}{r^2} (\boldsymbol{\Omega}^{(1)} \times \hat{\mathbf{r}}). \quad (3.10)$$

Successive steps follow as this result has been obtained. The solvent flow $\mathbf{u}^{(n)}$ induces translational and rotational velocities in the next chain, which consequently scatters the solvent yielding $\mathbf{u}^{(n+1)}$. The velocities of the successive chains are found by applying the zero-net-force and zero-net-torque conditions, so that a fluid flow $\mathbf{u}^{(n)}$ induces a translational velocity $\mathbf{V}^{(n+1)}$ and a rotational velocity $\boldsymbol{\Omega}^{(n+1)}$ on the $(n+1)$ st chain.

The force on a single bead due to $\mathbf{u}^{(1)}$, considering the motion of the entire chain as a result of the forces on all of its beads (neglecting internal mode relaxation) is

$$\mathbf{F}_i^{(2)} = f_i \left(\mathbf{u}^{(1)}(\mathbf{r}) - \mathbf{V}^{(2)} - \boldsymbol{\Omega}^{(2)} \times \mathbf{s}_i \right). \quad (3.11)$$

If $\mathbf{u}^{(1)}(\mathbf{r})$ is expanded in powers of $\mathbf{s} \cdot \nabla$ about $\mathbf{r} = \mathbf{R}_1$, the zero-net-force constraint is

$$f_0 \int dsg(s) \left[\mathbf{u}^{(1)}(\mathbf{R}_1) + (\mathbf{s} \cdot \nabla) \mathbf{u}^{(1)}(\mathbf{R}_1) - \mathbf{V}^{(2)} - \boldsymbol{\Omega}^{(2)} \times \mathbf{s} \right] = 0. \quad (3.12)$$

Eliminating terms that vanish by reflection symmetry for \mathbf{s} , equation 3.12 leads to

$$\mathbf{V}^{(2)} = \mathbf{u}^{(1)}(\mathbf{R}_1). \quad (3.13)$$

Iterating the argument for different labels in equation 3.12 leads to a generalization of equation 3.13, or, to lowest order in \mathbf{s}

$$\mathbf{V}^{(n+1)} = \mathbf{u}^{(n)}(\mathbf{R}_n). \quad (3.14)$$

The zero-net-torque constraint leads to an equation for the angular velocity of polymer 2.

$$\sum_{i=1}^M f_i \left[\mathbf{s}_i \times \mathbf{u}^{(1)}(\mathbf{r}_i) - \mathbf{s}_i \times \mathbf{V}^{(2)} - \mathbf{s}_i \times (\boldsymbol{\Omega}^{(2)} \times \mathbf{s}_i) \right] = 0. \quad (3.15)$$

The sum over a specific set of \mathbf{s}_i is then replaced by an average over the bead density;

terms odd in \mathbf{s} vanish because they integrate to zero. Hence

$$\frac{2}{3}f_0 \langle R_{g2}^2 \rangle \boldsymbol{\Omega}^{(2)} = f_0 \int ds g(s) \mathbf{s} \times (\mathbf{s} \cdot \nabla_{\mathbf{R}}) \mathbf{u}^{(1)}(\mathbf{R}). \quad (3.16)$$

Here the subscript on ∇ specifies the coordinate for its derivatives.

Applying the results $\mathbf{s} \cdot \nabla_{\mathbf{r}}(1/r^n) = -n\mathbf{s} \cdot \mathbf{r}/r^{n+2}$ and $\mathbf{s} \cdot \nabla_{\mathbf{r}}(\hat{\mathbf{r}}) = \mathbf{s} \cdot [\mathbf{I} - \hat{\mathbf{r}}\hat{\mathbf{r}}]/r$, and realizing that spherical averages over $g(s)$ lead to $\langle (\mathbf{s} \cdot \mathbf{a})(\mathbf{s} \times \mathbf{b}) \rangle = \frac{1}{3}R_g^2(\mathbf{a} \times \mathbf{b})$; $\langle (\mathbf{s} \cdot \mathbf{a})(\mathbf{s} \times (\mathbf{b} \times \mathbf{a})) \rangle = \frac{1}{3}R_g^2\mathbf{b} \times [\mathbf{I} - 3\mathbf{a}\mathbf{a}]$; $\langle \mathbf{s}(\mathbf{b} \cdot \mathbf{s}) \rangle = R_g^2\mathbf{b}/3$; and $\langle \mathbf{s} \times (\mathbf{b} \times \mathbf{s}) \rangle = \frac{2}{3}R_g^2\mathbf{b}$, equation 3.16 leads to

$$\boldsymbol{\Omega}^{(2)}(\mathbf{R}_1) = -\frac{3}{4} \frac{R_{h1}}{R_1^2} [\hat{\mathbf{R}}_1 \times \mathbf{V}^{(1)}] - \frac{1}{4} \frac{R_{h1}R_{g1}^2}{R_1^3} \boldsymbol{\Omega}^{(1)} \cdot [\mathbf{I} - 3\hat{\mathbf{R}}_1\hat{\mathbf{R}}_1] \quad (3.17)$$

for a polymer chain at \mathbf{R}_1 .

The physical picture for a second polymer hydrodynamically responding to a first polymer is presented in equation 3.7 for the solvent flow near polymer 2, and equations 3.13 and 3.17 for the translational and rotational responses of polymer 2 to this flow. It may be noted that, fundamentally, $\mathbf{V}^{(2)}$ and $\boldsymbol{\Omega}^{(2)}$ are different physically from $\mathbf{V}^{(1)}$ and $\boldsymbol{\Omega}^{(1)}$. The pair of velocities for polymer 1 are externally imposed; the pair of velocities for polymer 2 are a result of the solvent flow through polymer 2. Cross-coupling may also be noted in equations 3.13 and 3.17: although one or the other, but not both, of the translational and rotational velocities of polymer 1 are zero, polymer 2 has both non-zero translational and rotational velocities.

Next polymer 2 scatters the fluid flow $\mathbf{u}^{(1)}$ adding a flow $\mathbf{u}^{(2)}$. Utilizing equation 3.7 and recognizing that $\mathbf{u}^{(1)}$ must be subtracted from bead j 's velocity \mathbf{v}_j with respect to the stationary solvent, the net effect of polymer 2 is the vector sum of the contributions of the individual beads j as follows.

$$\mathbf{u}^{(2)}(\mathbf{r}) = \sum_{j=1}^M \mathbf{T}(\mathbf{r} - \mathbf{s}_j) \cdot f_j \left[\mathbf{v}_j(\mathbf{s}_j) - \mathbf{u}^{(1)}(\mathbf{R}_1 + \mathbf{s}_j) \right]. \quad (3.18)$$

Here \mathbf{r} and \mathbf{s} are measured relative to the center-of-mass of polymer 2.

Now \mathbf{s}_j is replaced by a variable \mathbf{s} , the position relative to the polymer center-of-mass; the bead configuration is replaced by a radial distribution function $g(s)$; and \mathbf{T} and \mathbf{u} are expanded in powers of $\mathbf{s} \cdot \nabla$, eliminating terms odd in \mathbf{s} . Invoking equations 3.6 and 3.13, it can be shown that generally

$$\mathbf{u}^{(n+1)}(\mathbf{r}) = \frac{f_0}{8\pi\eta} \int d\mathbf{s} \frac{g(s)}{r^2} \left[-\hat{\mathbf{r}}\mathbf{s} \cdot [\mathbf{I} - 3\hat{\mathbf{r}}\hat{\mathbf{r}}] - \mathbf{s}\hat{\mathbf{r}} + (\mathbf{s} \cdot \hat{\mathbf{r}})\mathbf{I} \right] \cdot \left[\Omega^{(n+1)} \times \mathbf{s} - \mathbf{s} \cdot \nabla [\mathbf{u}^{(n)}] \right]. \quad (3.19)$$

For averages over $g(s)$, the results $\langle (\mathbf{s} \cdot \mathbf{a})(\mathbf{s} \cdot \mathbf{b}) \rangle = \frac{1}{3}R_g^2 \mathbf{a} \cdot \mathbf{b}$, $\langle (\mathbf{s} \cdot \mathbf{a})(\mathbf{b} \times \mathbf{s}) \rangle = \frac{1}{3}R_g^2 \mathbf{b} \times \mathbf{a}$, and $\langle \mathbf{s}\mathbf{a} \cdot (\mathbf{b} \times \mathbf{s}) \rangle = \frac{1}{3}R_g^2 \mathbf{a} \times \mathbf{b}$ lead to

$$\begin{aligned} \mathbf{u}^{(n+1)}(\mathbf{r}) = \frac{R_{h,n+1}R_{g,n+1}^2}{4r^2} & \left[2(\Omega^{(n+1)} \times \hat{\mathbf{r}}) - 3\hat{\mathbf{r}}\hat{\mathbf{r}} \cdot [(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}})\mathbf{u}^{(n)}(\mathbf{R}_n)] \right. \\ & \left. + \nabla_{\mathbf{R}}(\hat{\mathbf{r}} \cdot \mathbf{u}^{(n)}(\mathbf{R}_n)) - \hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}}(\mathbf{u}^{(n)}(\mathbf{R}_n)) \right]. \quad (3.20) \end{aligned}$$

where $\nabla_{\mathbf{R}}$ is the gradient with respect to the highest numbered of the variables of the function being acted upon.

To obtain $\mathbf{V}^{(3)}$, equation 3.14 is applied.

$$\mathbf{V}^{(3)} = \mathbf{u}^{(2)}(\mathbf{R}_2). \quad (3.21)$$

The zero-net-torque constraint for the n th scattering occurrence is applied to give $\mathbf{\Omega}^{(n+1)}$. This constraint is

$$\sum_{i=1}^M f_i \left[\mathbf{s}_i \times \mathbf{u}^{(n)}(\mathbf{r}_i) - \mathbf{s}_i \times \mathbf{V}^{(n+1)} - \mathbf{s}_i \times (\mathbf{\Omega}^{(n+1)} \times \mathbf{s}_i) \right] = 0. \quad (3.22)$$

Converting the sum over \mathbf{s}_i to an integral over \mathbf{s} , the position relative to the polymer center-of-mass, and utilizing the distribution function $g(s)$, to lowest order in \mathbf{s} yields

$$\frac{2}{3} R_{g,n+1}^2 \mathbf{\Omega}^{(n+1)} = \int ds g(s) \mathbf{s} \times (\mathbf{s} \cdot \nabla) \mathbf{u}^{(n)}(\mathbf{r}_n). \quad (3.23)$$

Expanding \mathbf{s} in Cartesian coordinates converts the above integral to an integral over s^2 and a constant which does not depend on s . Consequently

$$\mathbf{\Omega}^{(n+1)} = \frac{1}{2} (\nabla \times \mathbf{u}^{(n)}(\mathbf{R}_n)) \quad (3.24)$$

to order $O(s^2)$. Setting $n = 2$ in equation 3.24 and applying the differential identities $\nabla_r \times \hat{\mathbf{r}} = 0$, $\nabla_r \times [(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}})^2 \hat{\mathbf{r}} / r^3] = [2(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}})(\hat{\mathbf{R}} \times \hat{\mathbf{r}}) / r^3]$, $\nabla_r \times (\mathbf{r} \times \mathbf{a}) / r^3 = (\mathbf{a} / r^3) \cdot$

$[\mathbf{I} - 3\hat{\mathbf{r}}\hat{\mathbf{r}}]$, $\nabla_r \times (\mathbf{r} \cdot \mathbf{a})\mathbf{b}/r^3 = [\mathbf{a} \cdot (\mathbf{I} - 3\hat{\mathbf{r}}\hat{\mathbf{r}}) \times \mathbf{b}]/r^3$, and $\nabla_r \times [(\mathbf{a} \cdot \hat{\mathbf{r}})(\mathbf{b} \cdot \hat{\mathbf{r}})\hat{\mathbf{r}}/r^2] = (\mathbf{b} \cdot \hat{\mathbf{r}})\mathbf{a} \times \hat{\mathbf{r}}/r^3 + (\mathbf{a} \cdot \hat{\mathbf{r}})\mathbf{b} \times \hat{\mathbf{r}}/r^3$ leads to an angular velocity of polymer 3 of

$$\begin{aligned} \boldsymbol{\Omega}^{(3)} = & \frac{27 R_{h1} R_{h2} R_{g2}^2 \mathbf{R}_1 \cdot \mathbf{V}^{(1)}}{64 R_1^2 R_2^3} \hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_1 (\hat{\mathbf{R}}_1 \times \hat{\mathbf{R}}_2) + \\ & \frac{9 R_{h1} R_{h2} R_{g1}^2 R_{g2}^2}{16 R_1^3 R_2^3} \left[\boldsymbol{\Omega}^{(1)} (1 - 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_1)^2) \right. \\ & \left. - \hat{\mathbf{R}}_1 \boldsymbol{\Omega}^{(1)} \cdot [\mathbf{I} - 2\hat{\mathbf{R}}_2 \hat{\mathbf{R}}_1] \cdot \hat{\mathbf{R}}_1 - \hat{\mathbf{R}}_2 \boldsymbol{\Omega}^{(1)} \cdot [\mathbf{I} - \hat{\mathbf{R}}_1 \hat{\mathbf{R}}_1] \cdot \hat{\mathbf{R}}_2 \right]. \end{aligned} \quad (3.25)$$

Simplification of the recursive formula for $\mathbf{u}^{(n+1)}$ follows. Equation 3.20 provides a recursive relation for $\mathbf{u}^{(n+1)}$, the current induced by the $(n+1)$ st polymer due to its translation and rotation under the influence of the n th polymer. Equation 3.24 provides a relation for $\boldsymbol{\Omega}^{(n+1)}$, the rotational velocity of the $(n+1)$ st polymer due to the disturbance caused by the n th polymer, namely

$$\boldsymbol{\Omega}^{(n+1)} = \frac{1}{2} (\nabla \times \mathbf{u}^{(n)}(\mathbf{R}_n)). \quad (3.26)$$

We now show that three of the terms of equation 3.20 in $\mathbf{u}^{(n+1)}$ cancel, leaving only the second term in square brackets. This premise would imply that

$$2(\boldsymbol{\Omega}^{(n+1)} \times \hat{\mathbf{r}}) = -\nabla_{\mathbf{R}}(\hat{\mathbf{r}} \cdot \mathbf{u}^{(n)}(\mathbf{R}_n)) + \hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}}(\mathbf{u}^{(n)}(\mathbf{R}_n)) \quad (3.27)$$

Evaluating $\boldsymbol{\Omega}^{(n+1)}$ in Cartesian components yields

$$\nabla_{\mathbf{R}} \times \mathbf{u}^{(n)} = \hat{\mathbf{x}} \left[\frac{\partial u_z}{\partial y_n} - \frac{\partial u_y}{\partial z_n} \right] + \hat{\mathbf{y}} \left[\frac{\partial u_x}{\partial z_n} - \frac{\partial u_z}{\partial y_n} \right] + \hat{\mathbf{z}} \left[\frac{\partial u_y}{\partial x_n} - \frac{\partial u_x}{\partial y_n} \right] \quad (3.28)$$

Taking the vector product of $\boldsymbol{\Omega}^{(n+1)}$ with $\hat{\mathbf{r}}$ leads to

$$\begin{aligned} 2(\boldsymbol{\Omega}^{(n+1)} \times \hat{\mathbf{r}}) &= \hat{\mathbf{x}} \left[\frac{z_{n+1}}{r_{n+1}} \left(\frac{\partial u_x}{\partial z_n} - \frac{\partial u_z}{\partial x_n} \right) - \frac{y_{n+1}}{r_{n+1}} \left(\frac{\partial u_y}{\partial x_n} - \frac{\partial u_x}{\partial y_n} \right) \right] \\ &+ \hat{\mathbf{y}} \left[\frac{x_{n+1}}{r_{n+1}} \left(\frac{\partial u_y}{\partial x_n} - \frac{\partial u_x}{\partial y_n} \right) - \frac{z_{n+1}}{r_{n+1}} \left(\frac{\partial u_z}{\partial y_n} - \frac{\partial u_y}{\partial z_n} \right) \right] \\ &+ \hat{\mathbf{z}} \left[\frac{y_{n+1}}{r_{n+1}} \left(\frac{\partial u_z}{\partial y_n} - \frac{\partial u_y}{\partial z_n} \right) - \frac{x_{n+1}}{r_{n+1}} \left(\frac{\partial u_x}{\partial z_n} - \frac{\partial u_z}{\partial x_n} \right) \right] \end{aligned} \quad (3.29)$$

for the left-hand-side of equation 3.27.

Furthermore

$$\hat{\mathbf{r}} \cdot \mathbf{u}^{(n)} = [x_{n+1}u_{x,n} + y_{n+1}u_{y,n} + z_{n+1}u_{z,n}]/r_{n+1} \quad (3.30)$$

The gradient with respect to \mathbf{R}_n of this scalar product is

$$\begin{aligned} \nabla_{\mathbf{R}_n} (\hat{\mathbf{r}} \cdot \mathbf{u}^{(n)}) &= \frac{x_{n+1}}{r_{n+1}} \left[\frac{\partial u_x}{\partial x_n} \hat{\mathbf{x}} + \frac{\partial u_x}{\partial y_n} \hat{\mathbf{y}} + \frac{\partial u_x}{\partial z_n} \hat{\mathbf{z}} \right] \\ &+ \frac{y_{n+1}}{r_{n+1}} \left[\frac{\partial u_y}{\partial x_n} \hat{\mathbf{x}} + \frac{\partial u_y}{\partial y_n} \hat{\mathbf{y}} + \frac{\partial u_y}{\partial z_n} \hat{\mathbf{z}} \right] \\ &+ \frac{z_{n+1}}{r_{n+1}} \left[\frac{\partial u_z}{\partial x_n} \hat{\mathbf{x}} + \frac{\partial u_z}{\partial y_n} \hat{\mathbf{y}} + \frac{\partial u_z}{\partial z_n} \hat{\mathbf{z}} \right] \end{aligned} \quad (3.31)$$

In addition,

$$\begin{aligned}
(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}_n}) \mathbf{u}^{(n)} &= \frac{x_{n+1}}{r_{n+1}} \left[\frac{\partial u_x}{\partial x_n} \hat{\mathbf{x}} + \frac{\partial u_y}{\partial x_n} \hat{\mathbf{y}} + \frac{\partial u_z}{\partial x_n} \hat{\mathbf{z}} \right] \\
&+ \frac{y_{n+1}}{r_{n+1}} \left[\frac{\partial u_x}{\partial y_n} \hat{\mathbf{x}} + \frac{\partial u_y}{\partial y_n} \hat{\mathbf{y}} + \frac{\partial u_z}{\partial y_n} \hat{\mathbf{z}} \right] \\
&+ \frac{z_{n+1}}{r_{n+1}} \left[\frac{\partial u_x}{\partial z_n} \hat{\mathbf{x}} + \frac{\partial u_y}{\partial z_n} \hat{\mathbf{y}} + \frac{\partial u_z}{\partial z_n} \hat{\mathbf{z}} \right]
\end{aligned} \tag{3.32}$$

Combining equation 3.31 with equation 3.32 leads to

$$\begin{aligned}
-\nabla_{\mathbf{R}_n}(\hat{\mathbf{r}} \cdot \mathbf{u}^{(n)}) + (\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}_n}) \mathbf{u}^{(n)} &= \hat{\mathbf{x}} \left[\frac{x_{n+1}}{r_{n+1}} \left(\frac{\partial u_x}{\partial x_n} - \frac{\partial u_x}{\partial x_n} \right) + \frac{y_{n+1}}{r_{n+1}} \left(\frac{\partial u_x}{\partial y_n} - \frac{\partial u_y}{\partial x_n} \right) \right. \\
&\quad \left. + \frac{z_{n+1}}{r_{n+1}} \left(\frac{\partial u_x}{\partial z_n} - \frac{\partial u_z}{\partial x_n} \right) \right] \\
&+ \hat{\mathbf{y}} \left[\frac{x_{n+1}}{r_{n+1}} \left(\frac{\partial u_y}{\partial x_n} - \frac{\partial u_x}{\partial y_n} \right) + \frac{y_{n+1}}{r_{n+1}} \left(\frac{\partial u_y}{\partial y_n} - \frac{\partial u_y}{\partial y_n} \right) \right. \\
&\quad \left. + \frac{z_{n+1}}{r_{n+1}} \left(\frac{\partial u_y}{\partial z_n} - \frac{\partial u_z}{\partial y_n} \right) \right] \\
&+ \hat{\mathbf{z}} \left[\frac{x_{n+1}}{r_{n+1}} \left(\frac{\partial u_z}{\partial x_n} - \frac{\partial u_x}{\partial z_n} \right) + \frac{y_{n+1}}{r_{n+1}} \left(\frac{\partial u_z}{\partial y_n} - \frac{\partial u_y}{\partial z_n} \right) \right. \\
&\quad \left. + \frac{z_{n+1}}{r_{n+1}} \left(\frac{\partial u_z}{\partial z_n} - \frac{\partial u_z}{\partial z_n} \right) \right] \tag{3.33}
\end{aligned}$$

Eliminating terms which add to zero, and combining equations 3.33 and 3.29,

$$2(\boldsymbol{\Omega}^{(n+1)} \times \hat{\mathbf{r}}) + \nabla_{\mathbf{R}}(\hat{\mathbf{r}} \cdot \mathbf{u}^{(n)}(\mathbf{R}_n)) + \hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}}(\mathbf{u}^{(n)}(\mathbf{R}_n)) = 0 \tag{3.34}$$

Hence, for general n ,

$$\mathbf{u}^{(n+1)} = -\frac{R_{h,n+1}R_{g,n+1}^2}{4r^2}3\hat{\mathbf{r}}\hat{\mathbf{r}} \cdot [(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}})\mathbf{u}^{(n)}(\mathbf{R}_n)] \quad (3.35)$$

Letting $n = 1$, utilizing $\boldsymbol{\Omega}^{(2)}$ from equation 3.17, and applying the above differential identities to $\mathbf{r} \cdot \nabla_{\mathbf{R}}\hat{\mathbf{R}}$ and $\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}}R^{-m}$ leads to

$$\begin{aligned} \mathbf{u}^{(2)}(\mathbf{r}) = & -\frac{9}{16}\frac{R_{h1}R_{h2}R_{g2}^2}{R_1^2r^2}\left[1 - 3(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_1)^2\right]\hat{\mathbf{r}}(\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)}) \\ & +\frac{3}{8}\frac{R_{h1}^2R_{h2}^2R_{g1}^2R_{g2}^2}{R_1^3r^2}\left[\hat{\mathbf{r}} \times \boldsymbol{\Omega}^{(1)} - (\hat{\mathbf{r}} \times \hat{\mathbf{R}}_1)\hat{\mathbf{R}}_1 \cdot \boldsymbol{\Omega}^{(1)} \right. \\ & \left. +\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_1(\boldsymbol{\Omega}^{(1)} \times \hat{\mathbf{R}}_1) - \hat{\mathbf{r}} \cdot (\boldsymbol{\Omega}^{(1)} \times \hat{\mathbf{R}}_1)\hat{\mathbf{R}}_1 \cdot (\mathbf{I} - 3\hat{\mathbf{r}}\hat{\mathbf{r}})\right] \end{aligned} \quad (3.36)$$

for the flow hydrodynamically scattered by polymer 2.

Having found $\mathbf{V}^{(3)}$ and $\boldsymbol{\Omega}^{(3)}$, Phillis then calculates[1] $\mathbf{u}^{(3)}$ from equation 3.20, arriving at

$$\begin{aligned} \mathbf{u}^{(3)}(\mathbf{r}) = & \frac{27}{64}\frac{R_{h1}R_{h2}R_{h3}R_{g2}^2R_{g3}^2}{R_1^2R_2^3r^2}\left[\left(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2\right)\left(1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{r}})^2\right) \right. \\ & \left. -6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{r}})\hat{\mathbf{r}} \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1\right](\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)})\hat{\mathbf{r}}. \end{aligned} \quad (3.37)$$

From equation 3.14

$$\mathbf{V}^{(4)} = \mathbf{u}^{(3)}(\mathbf{R}_3). \quad (3.38)$$

Phillis[1] then derives \mathbf{T}_{21} , \mathbf{T}_{321} , \mathbf{T}_{4321} , \mathbf{b}_{12} , and \mathbf{b}_{123} , the cross- and self-interaction tensors from equations 3.3, 3.4, and 3.5 and the relations for $\mathbf{V}^{(n)}$ and $\boldsymbol{\Omega}^{(n)}$. Specifically, \mathbf{T}_{ij} , \mathbf{T}_{ilj} , and \mathbf{T}_{ilmj} are derived from the terms in $\mathbf{V}^{(2)}$, $\mathbf{V}^{(3)}$, and

$\mathbf{V}^{(4)}$ which are proportional to $\mathbf{V}^{(1)}$, respectively, after demanding that the first and last particles of a given scattering sequence be distinct, by factoring $f_{01}\mathbf{V}^{(1)}$ out of $\mathbf{V}^{(n)}$. From equation 3.10

$$\mathbf{T}_{21} = \frac{1}{f_{01}} \frac{3 R_{h1}}{4 R_1} [\mathbf{I} + \hat{\mathbf{R}}_1 \hat{\mathbf{R}}_1]. \quad (3.39)$$

From equation 3.36 it results that

$$\mathbf{T}_{321} = -\frac{1}{f_{01}} \frac{9 R_{h1} R_{h2} R_{g2}^2}{16 R_1^2 R_2^2} [1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_1)^2] \hat{\mathbf{R}}_2 \hat{\mathbf{R}}_1. \quad (3.40)$$

Finally, from equation 3.37

$$\begin{aligned} \mathbf{T}_{4321} = \frac{1}{f_{01}} \frac{27 R_{h1} R_{h2} R_{h3} R_{g2}^2 R_{g3}^2}{64 R_1^2 R_2^3 R_3^2} & \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2) (1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2) \right. \\ & \left. - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) \hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2 \hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \right] \hat{\mathbf{R}}_3 \hat{\mathbf{R}}_1. \end{aligned} \quad (3.41)$$

The self-interaction tensors \mathbf{b}_{il} and \mathbf{b}_{ikl} are derived[1] from $\mathbf{V}^{(3)}$ and $\mathbf{V}^{(4)}$, respectively, with the constraint that the first and last polymers in a scattering sequence be the same, and factoring out an $f_{01}\mathbf{V}^{(1)}$. The result for \mathbf{b}_{12} is

$$\mathbf{b}_{12} = -\frac{1}{f_{01}} \frac{9 R_{h1} R_{h2} R_{g2}^2}{8 R_1^4} \hat{\mathbf{R}}_1 \hat{\mathbf{R}}_1 \quad (3.42)$$

while Phillies' result[1] for the three polymer ring is

$$\mathbf{b}_{321} = \mathbf{T}_{4321}|_{(\mathbf{R}_3 = -\mathbf{R}_2 - \mathbf{R}_1)}. \quad (3.43)$$

3.2 Ensemble Average of the Hydrodynamic Interaction Tensors

In this section Phillies' determination of the leading coefficients B_2 and B_3 of the pseudovirial expansion of the polymer self-diffusion coefficient[1] due to rings

$$D_s \approx D_0 (1 + B_2 c + B_3 c^2) \quad (3.44)$$

is presented.

A generalized Einstein relation provides an expression for the self-diffusion coefficient.

$$D_s = \frac{k_B T}{3} \text{Tr} \frac{1}{N} \left\langle \sum_{i=1}^N \mu_{ii} \right\rangle. \quad (3.45)$$

Here the trace is taken with respect to the Cartesian coordinates; the brackets $\langle \dots \rangle$ signify an ensemble average. The mobility μ_{ii} expansion is

$$\mu_{ii} = \frac{\mathbf{I}}{f_{0i}} + \sum_{l=1}^N \mathbf{b}_{il} + \sum_{l,m=1}^N \mathbf{b}_{ilm} + \dots. \quad (3.46)$$

If two or more of the indices of the \mathbf{b} 's are equal, the corresponding \mathbf{b} is zero.

In order to take the ensemble average in equation 3.45, the N -particle potential energy U_N and the polymers' center-of-mass coordinates $\mathbf{r}_1, \mathbf{r}_2, \dots$, and relative coordinates $\mathbf{r}_{ij} = \mathbf{r}_j - \mathbf{r}_i$ are invoked. The n -polymer distribution functions $g^{(n)}(\mathbf{r}_{12}, \dots, \mathbf{r}_{1n})$ are given, for example, by

$$cg^{(2)}(\mathbf{r}_{12}) = \frac{N \int d\mathbf{r}_{13} \dots d\mathbf{r}_{1N} \exp(-\beta U_N)}{\int d\mathbf{r}_{12} \dots d\mathbf{r}_{1N} \exp(-\beta U_N)}. \quad (3.47)$$

Here $c = N/V$ is the polymer number concentration. The self-diffusion coefficient may be written in terms of the $g^{(n)}(\mathbf{r}_{12}, \dots, \mathbf{r}_{1n})$ and the μ_{ii} as

$$D_s = \frac{k_B T}{f_{01}} \left[1 + \frac{f_0}{3} \left\{ c \int_V d\mathbf{r}_{12} \mathbf{b}_{12} g^{(2)}(\mathbf{r}_{12}) + c^2 \int_V d\mathbf{r}_{12} d\mathbf{r}_{13} \mathbf{b}_{123} g^{(3)}(\mathbf{r}_{12}, \mathbf{r}_{13}) \right\} \right]. \quad (3.48)$$

Equating like powers in equation 3.44 and 3.45, and utilizing equation 3.42 for \mathbf{b}_{12} , an expression for B_2

$$B_2 = \frac{1}{3} \left\{ c \int_V d\mathbf{r}_{12} \left(-\frac{9}{8} \frac{R_{h1} R_{h2} R_{g2}^2}{r_{12}^4} \hat{\mathbf{r}}_{12} \hat{\mathbf{r}}_{12} \right) g^{(2)}(\mathbf{r}_{12}) \right\} \quad (3.49)$$

follows. In order to evaluate B_2 , Phillies[1] assumed a form for $g^{(2)}(\mathbf{r}_{12})$. He took this distribution function to be flat subject to the constraint that $r_{12} \geq 2a_0$, where $2a_0$ is the bead-bead distance of closest approach.

Straightforward integration of equation 3.49 yields

$$B_2 = -\frac{9}{16} \frac{R_{h1}R_{h2}}{a_0R_g} \left(\frac{4\pi}{3} R_g^3 \right). \quad (3.50)$$

In a manner similar to that utilized to obtain 3.49, Phillies[1] generates an expression for B_3 , namely

$$B_3 = \frac{9}{64} \frac{R_{h1}R_{h2}R_{h3}R_{g2}^2R_{g3}^2}{a_0} \left\{ \int_2^\infty r^2 dr \int_2^\infty s^2 ds \int_{4\pi} d\Omega_r d\Omega_s \frac{g^{(2)}(r_{23})}{r^2 r_{23}^3 s^2} \text{Tr} [f(\Omega_r, \Omega_s)] \right\} \quad (3.51)$$

Here r , s , and r_{23} are dimensionless distances in units of $a_0 = 1$, where r is the distance from polymer 1 to polymer 2, s is the distance from polymer 1 to polymer 3, and r_{23} is the distance from polymer 2 to polymer 3. The purpose of $g^{(2)}(r_{23})$ is to ensure the excluded volume between polymers 2 and 3. The argument of the trace is the bracketed function of unit vectors in equation 3.41 for \mathbf{T}_{4321} subject to the constraint that $\mathbf{R}_3 = -\mathbf{R}_1 - \mathbf{R}_2$. A numerical value for B_3 was then determined using a Monte Carlo routine, finding

$$B_3 = 9.321 \times 10^{-4} \frac{R_{h1}R_{h2}R_{h3}}{a_0R_{g2}R_{g3}} \frac{4\pi R_{g2}^3}{3} \frac{4\pi R_{g3}^3}{3}. \quad (3.52)$$

Chapter 4

Derivation of $\mathbf{u}^{(4)}$

4.1 Analysis

$\mathbf{u}^{(n+1)}$ as given in equation 3.35 is

$$\mathbf{u}^{(n+1)} = -\frac{R_{h,n+1}R_{g,n+1}^2}{4r^2}3\hat{\mathbf{r}} \cdot [(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}})\mathbf{u}^{(n)}(\mathbf{R}_n)]\hat{\mathbf{r}} \quad (4.1)$$

This equation will be employed to determine $\mathbf{u}^{(4)}$ from $\mathbf{u}^{(3)}$. In equation 4.1, if $n = 3$, \mathbf{r} is the point in the solvent relative to the fourth polymer. $\mathbf{u}^{(3)}$, the fluid flow at a point \mathbf{r} in the solvent relative to polymer 3 due to scattering of $\mathbf{u}^{(2)}$ by the third polymer, is

$$\begin{aligned} \mathbf{u}^{(3)}(\mathbf{r}) = \frac{27}{64} \frac{R_{h1}R_{h2}R_{h3}R_{g2}^2R_{g3}^2}{R_1^2R_2^3r^2} & \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2) (1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{r}})^2) \right. \\ & \left. - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{r}})\hat{\mathbf{r}} \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \right] (\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)})\hat{\mathbf{r}}. \end{aligned} \quad (4.2)$$

Since $\mathbf{u}^{(3)}(\mathbf{r})$ is used to determine $\mathbf{u}^{(4)}$ in equation 4.1, one needs $\mathbf{u}^{(3)}(\mathbf{R}_3)$, which is

$$\begin{aligned} \mathbf{u}^{(3)}(\mathbf{R}_3) = & \frac{27}{64} \frac{R_{h1} R_{h2} R_{h3} R_{g2}^2 R_{g3}^2}{R_1^2 R_2^3 R_3^2} \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2) (1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2) \right. \\ & \left. - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) \hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2 \hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \right] (\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)}) \hat{\mathbf{R}}_3. \end{aligned} \quad (4.3)$$

To evaluate equation 4.1, $(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}}) \mathbf{u}^{(n)}(\mathbf{R}_n)$ is expanded as

$$(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}}) \mathbf{u}^{(3)}(\mathbf{R}_3) = \left[\frac{x}{r} \frac{\partial}{\partial x_3} \mathbf{u}^{(3)} + \frac{y}{r} \frac{\partial}{\partial y_3} \mathbf{u}^{(3)} + \frac{z}{r} \frac{\partial}{\partial z_3} \mathbf{u}^{(3)} \right] \quad (4.4)$$

Define A by

$$A = \frac{27}{64} \frac{a}{R_3^3} \left[(1 - 3b^2)(1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2) + 6b(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) \left((\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_3) - b(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) \right) \right]. \quad (4.5)$$

Here $a = (R_{h1} R_{h2} R_{h3} R_{g2}^2 R_{g3}^2) / (R_1^2 R_2^3)$ and $b = (\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2$.

From equation 4.3, $\mathbf{u}^{(3)}$ may be written as

$$\mathbf{u}^{(3)} = A(\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)})(x_3 \hat{\mathbf{x}} + y_3 \hat{\mathbf{y}} + z_3 \hat{\mathbf{z}}), \quad (4.6)$$

where $\mathbf{R}_3 = (x_3 \hat{\mathbf{x}} + y_3 \hat{\mathbf{y}} + z_3 \hat{\mathbf{z}})$. Writing \mathbf{R}_3 in components everywhere, A becomes

$$A = \frac{27}{64} \frac{a}{(x_3^2 + y_3^2 + z_3^2)^{3/2}} \left[(1 - 3b^2) \left(1 - 3 \frac{(x_2 x_3 + y_2 y_3 + z_2 z_3)^2}{R_2^2 (x_3^2 + y_3^2 + z_3^2)} \right) \right]$$

$$-6b \frac{(x_2x_3 + y_2y_3 + z_2z_3)}{R_2(x_3^2 + y_3^2 + z_3^2)} \left(\frac{x_1x_3 + y_1y_3 + z_1z_3}{R_1} - b \frac{(x_2x_3 + y_2y_3 + z_2z_3)}{R_2} \right) \Big]. \quad (4.7)$$

In equations 4.6 and 4.7, x_3 , y_3 , and z_3 appear on equal footing, which will be useful in taking the gradient of A with respect to \mathbf{R}_3 .

Substituting equation 4.6 into equation 4.4, one finds

$$x \frac{\partial}{\partial x_3} \mathbf{u}^{(3)} = x (\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)}) \left[\frac{\partial A}{\partial x_3} \mathbf{R}_3 + A \hat{\mathbf{x}} \right] \quad (4.8)$$

and similarly for the terms $\partial/\partial y_3$ and $\partial/\partial z_3$. Again, $\mathbf{r} = (x\hat{\mathbf{x}} + y\hat{\mathbf{y}} + z\hat{\mathbf{z}})$ is the position in the solvent relative to the fourth polymer, From equations 4.4 and 4.8, one obtains

$$(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}}) \mathbf{u}^{(3)}(\mathbf{R}_3) = (\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)}) [(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}} A) \mathbf{R}_3 + A \hat{\mathbf{r}}] \quad (4.9)$$

To find $\nabla_{\mathbf{R}} A$, equation 4.7 is differentiated with respect to the components of \mathbf{R}_3 . The x_3 derivative is

$$\begin{aligned} \frac{\partial A}{\partial x_3} = & -\frac{3}{2} \frac{27}{64} \frac{2ax_3}{(x_3^2 + y_3^2 + z_3^2)^{5/2}} \left[(1 - 3b^2)(1 - 3c^2) - 6bc(d - bc) \right] \\ & + \frac{27}{64} \frac{a}{R_3^3} \left[(1 - 3b^2) \left[-6 \frac{x_2(x_2x_3 + y_2y_3 + z_2z_3)}{R_2^2 R_3^2} - 3 \frac{-2x_3c^2}{(x_3^2 + y_3^2 + z_3^2)} \right] \right. \\ & \left. - \frac{6bx_2}{R_2(x_3^2 + y_3^2 + z_3^2)^{1/2}} (d - bc) - \frac{-6b(x_2x_3 + y_2y_3 + z_2z_3)(2x_3)}{R_2(x_3^2 + y_3^2 + z_3^2)^{3/2}} (d - bc) \right] \end{aligned}$$

$$-\frac{6bc}{R_3} \left(\frac{x_1}{R_1} - \frac{bx_2}{R_2} \right) \quad (4.10)$$

Here $c = (\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)$ and $d = (\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_3)$. Using the a , b , c , and d symbols where applicable in the above equation, one obtains

$$\begin{aligned} \frac{\partial A}{\partial x_3} = & -\frac{81}{64} \frac{ax_3}{R_3^5} \left[(1 - 3b^2)(1 - 3c^2) - 6bc(d - bc) \right] + \frac{27}{64} \frac{a}{R_3^3} \left[(1 - 3b^2) \left[\frac{-6x_2c}{R_2R_3} \right. \right. \\ & \left. \left. + \frac{6x_3c^2}{R_3^2} \right] - \frac{6bx_2}{R_2R_3} (d - bc) + \frac{12bcx_3}{R_3^2} (d - bc) - \frac{6bc}{R_3} \left(\frac{x_1}{R_1} - \frac{bx_2}{R_2} \right) \right] \quad (4.11) \end{aligned}$$

From equation 4.11 and similar results for $\partial A/\partial y_3$ and $\partial A/\partial z_3$,

$$\begin{aligned} \hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}} A = & -\frac{81a}{64} \frac{\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_3}{R_3^4} \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2) - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2) \right. \\ & \times (\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) \hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2 \hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \left. \right] + \frac{27}{64} \frac{a}{R_3^4} \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2) \left[-6(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_2) \right. \right. \\ & \times (\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) + 6(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_3)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2 \left. \right] - 6(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2) \hat{\mathbf{R}}_1 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2 \hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_3 \\ & + 12(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_3)(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) \hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2 \hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \\ & \left. - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) \hat{\mathbf{r}} \cdot [\mathbf{I} - \hat{\mathbf{R}}_2 \hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \right] \quad (4.12) \end{aligned}$$

Factoring a -3 out of each subterm in the second term of the above equation, and combining like terms gives

$$\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}} A = -\frac{81}{64} \frac{a}{R_3^4} \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2)(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_3) - 10(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2) \right]$$

$$\begin{aligned}
& \times (\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 + 2(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) \\
& \quad \times \hat{\mathbf{r}} \cdot [\mathbf{I} - \hat{\mathbf{R}}_3\hat{\mathbf{R}}_3] \cdot \hat{\mathbf{R}}_2 + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_2)\hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \\
& \quad + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{r}} \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \Big\} (4.13)
\end{aligned}$$

Combining equations 4.1 and 4.9 results in

$$\mathbf{u}^{(4)}(\mathbf{r}) = -3 \frac{R_{h4} R_{g4}^2}{4r^2} \hat{\mathbf{r}} (\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)}) \hat{\mathbf{r}} \cdot [(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}}) A \mathbf{R}_3 + A \hat{\mathbf{r}}]. \quad (4.14)$$

Replacing $(\hat{\mathbf{r}} \cdot \nabla_{\mathbf{R}})A$ in equation 4.14 with equation 4.13 yields

$$\begin{aligned}
\mathbf{u}^{(4)}(\mathbf{r}) = & -3 \frac{R_{h4} R_{g4}^2}{4r^2} \hat{\mathbf{r}} (\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)}) \left\{ A + \hat{\mathbf{R}}_3 \cdot \hat{\mathbf{r}} \left[-\frac{81}{64} \frac{a}{R_3^3} \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2) \right. \right. \right. \\
& \times (1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2)(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_3) - 10(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2) \\
& (\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 + 2(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{r}} \cdot [\mathbf{I} - \hat{\mathbf{R}}_3\hat{\mathbf{R}}_3] \cdot \hat{\mathbf{R}}_2 \\
& \left. \left. \left. + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{r}} \cdot \hat{\mathbf{R}}_2)\hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \right] \right\} \quad (4.15)
\end{aligned}$$

Replacing A with its value as given in equation 4.7 yields

$$\begin{aligned}
\mathbf{u}^{(4)}(\mathbf{r}) = & -\frac{3R_{h4}R_{g4}^2}{4r^2} \frac{81}{64} \frac{R_{h1}R_{h2}R_{h3}R_{g2}^2R_{g3}^2}{R_1^2R_2^3R_3^3} \hat{\mathbf{r}} (\hat{\mathbf{R}}_1 \cdot \mathbf{V}^{(1)}) \left\{ (\hat{\mathbf{R}}_3 \cdot \hat{\mathbf{r}}) \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2) \right. \right. \\
& \times (1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2)(\hat{\mathbf{R}}_3 \cdot \hat{\mathbf{r}}) - 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)[5(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3) - (\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{r}})]\hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \\
& \left. \left. + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{r}} \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 + 2(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{r}} \cdot [\mathbf{I} - \hat{\mathbf{R}}_3\hat{\mathbf{R}}_3] \cdot \hat{\mathbf{R}}_2 \right] \right\}
\end{aligned}$$

$$+\frac{1}{3}[(1-3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1-3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2) - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1] \quad (4.16)$$

as a final expression for $\mathbf{u}^{(4)}(\mathbf{r})$, the velocity of the solvent due to quartets of polymers interacting hydrodynamically.

4.2 Numerical Confirmation of $\mathbf{u}^{(4)}$

Two programs which used pseudorandomly generated values for \mathbf{R}_1 , \mathbf{R}_2 , \mathbf{R}_3 , and \mathbf{R}_4 were run to confirm numerically equation 4.16, the algebraic form for $\mathbf{u}^{(4)}$, namely `polyalg.for` (see appendix A) and `hydrpoly.for` (see appendix B). `polyalg.for` computes $\mathbf{u}^{(4)}$ from equation 4.16. `hydrpoly.for` uses equation 3.20, the recursive relation for $\mathbf{u}^{(n)}$, and performs a numerical differentiation of $\mathbf{u}^{(3)}$ to determine $\mathbf{u}^{(4)}$. The two programs were run for 100 sets of 4 random points, and their predictions were compared. For a differential of ≈ 0.001 times the average length of a component of an \mathbf{R} , the analytic form of equation 3.20 and the numerical form computed using 4.16 agree in 98 % of the cases to one part in 10^4 .

Chapter 5

Evaluation of \mathbf{T}_{54321} and \mathbf{b}_{1232}

Factoring an $f_0 \mathbf{V}^{(1)}$ out of $\mathbf{u}^{(4)}$ and setting $\mathbf{r} = \mathbf{R}_4$, \mathbf{R}_4 being the position of the fifth polymer relative to the fourth polymer, one obtains \mathbf{T}_{54321} , the hydrodynamical cross-interaction tensor for quintets of polymers in solution.

$$\begin{aligned}
 \mathbf{T}_{54321} = & -\frac{3}{4f_0} \frac{R_{h1}R_{h2}R_{h3}R_{h4}R_{g2}^2R_{g3}^2R_{g4}^2}{R_1^2R_2^3R_3^3R_4^2} \frac{27}{64} \left\{ (1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2) \right. \\
 & - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \\
 & - 3(\hat{\mathbf{R}}_3 \cdot \hat{\mathbf{R}}_4) \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1 - 3(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)^2)(\hat{\mathbf{R}}_3 \cdot \hat{\mathbf{R}}_4) \right. \\
 & + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)\hat{\mathbf{R}}_3 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1\hat{\mathbf{R}}_4 \cdot [\mathbf{I} - 5\hat{\mathbf{R}}_3\hat{\mathbf{R}}_3] \cdot \hat{\mathbf{R}}_2 \\
 & + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{R}}_4 \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \\
 & \left. \left. + 2(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(\hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_3)\hat{\mathbf{R}}_4 \cdot [\mathbf{I} - \hat{\mathbf{R}}_3\hat{\mathbf{R}}_3] \cdot \hat{\mathbf{R}}_2 \right] \right\} \hat{\mathbf{R}}_4\hat{\mathbf{R}}_1 \quad (5.1)
 \end{aligned}$$

In equation 5.1, setting $\mathbf{R}_4 = -\mathbf{R}_1$, $\mathbf{R}_3 = -\mathbf{R}_2$, $R_{h4} = R_{h2}$, and $R_{g4} = R_{g2}$ gives

$\mathbf{b}_{1232} = \mathbf{T}_{54321}|_{\mathbf{R}_4=-\mathbf{R}_1; \mathbf{R}_3=-\mathbf{R}_2}$ the self-interaction tensor from a multiple scattering picture for a figure-eight scattering diagram,

$$\begin{aligned} \mathbf{b}_{1232} = & -\frac{3}{4f_0} \frac{R_{h1}R_{h2}^2R_{h3}R_{g2}^4R_{g3}^2}{R_1^4R_2^6} \frac{1}{R_1^2} (-\hat{\mathbf{R}}_1)(\hat{\mathbf{R}}_1) \frac{27}{64} \frac{1}{R_1^2R_2^6} \left\{ (1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2) \right. \\ & \times (1 - 3(\hat{\mathbf{R}}_2 \cdot (-\hat{\mathbf{R}}_2))^2) - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot (-\hat{\mathbf{R}}_2))(-\hat{\mathbf{R}}_2) \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 - 3((-\hat{\mathbf{R}}_2) \cdot (-\hat{\mathbf{R}}_1)) \\ & \times \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1 - 3(\hat{\mathbf{R}}_2 \cdot (-\hat{\mathbf{R}}_2))^2)(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2) + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(-\hat{\mathbf{R}}_2) \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \right. \\ & \times (-\hat{\mathbf{R}}_1) \cdot [\mathbf{I} - 5(-\hat{\mathbf{R}}_2)(-\hat{\mathbf{R}}_2)] \cdot \hat{\mathbf{R}}_2 + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)(\hat{\mathbf{R}}_2 \cdot (-\hat{\mathbf{R}}_2))(-\hat{\mathbf{R}}_1) \cdot [\mathbf{I} - \hat{\mathbf{R}}_2\hat{\mathbf{R}}_2] \cdot \hat{\mathbf{R}}_1 \\ & \left. \left. + 2(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(\hat{\mathbf{R}}_2 \cdot (-\hat{\mathbf{R}}_2))(-\hat{\mathbf{R}}_1) \cdot [\mathbf{I} - (-\hat{\mathbf{R}}_2)(-\hat{\mathbf{R}}_2)] \cdot \hat{\mathbf{R}}_2 \right\} \end{aligned} \quad (5.2)$$

Dropping terms which add to zero and evaluating the accessible scalar products,

$$\begin{aligned} \mathbf{b}_{1232} = & \frac{3}{4f_0} \frac{R_{h1}R_{h2}^2R_{h3}R_{g2}^4R_{g3}^2}{R_1^4R_2^6} \frac{27}{64} \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1 - 3) - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2) \right. \\ & \times \left. \left[(1 - 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2)(1 - 3)(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}) + 2(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}})(-1)(-1)(1 - (\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2) \right] \right] \hat{\mathbf{R}}_1\hat{\mathbf{R}}_1 \end{aligned} \quad (5.3)$$

Performing the arithmetic,

$$\mathbf{b}_{1232} = \frac{81}{128f_0} \frac{R_{h1}R_{h2}^2R_{h3}R_{g2}^4R_{g3}^2}{R_1^4R_2^6} [-1 + 3(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2 - 6(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^4] \hat{\mathbf{R}}_1\hat{\mathbf{R}}_1, \quad (5.4)$$

a final expression for \mathbf{b}_{1232} , the self-interaction tensor from a multiple scattering picture for the figure-eight scattering diagram.

Chapter 6

Ensemble Average of the Self-Interaction Tensor for the Figure-Eight Scattering Diagram

6.1 Analysis

Extending Phillies'[1] equation 3.48 for the self-diffusion coefficient one obtains a c^2 correction, namely

$$B'_3 c^2 = \frac{f_0}{3} c^2 \int_V d\mathbf{r}_{12} d\mathbf{r}_{13} \mathbf{b}_{1232} g^{(3)}(\mathbf{r}_{12}, \mathbf{r}_{13}). \quad (6.1)$$

This term involves an ensemble average over \mathbf{b}_{1232} . Evaluating 6.1 with equation 5.4 for \mathbf{b}_{1232} , using Phillies'[1] distribution function, gives

$$\begin{aligned} \langle \mathbf{b}_{1232} \rangle = & \frac{81}{128f_0} R_{h1} R_{h2}^2 R_{h3} R_{g2}^4 R_{g3}^2 \int_{2a_0}^{\infty} \frac{R_1^2 dR_1}{R_1^4} \int_{2a_0}^{\infty} \frac{R_2^2 dR_2}{R_2^6} \int_{4\pi} d\Omega_{R_1} \\ & \times \int d\phi \int_{-1}^1 d(\cos \theta) (-1 + 3 \cos^2 \theta - 6 \cos^4 \theta) \text{Tr} \langle \hat{\mathbf{R}}_1 \hat{\mathbf{R}}_1 \rangle \end{aligned} \quad (6.2)$$

Here θ is the angle between \mathbf{R}_1 and \mathbf{R}_2 . The lower bound on the R_1 and R_2 integrals appears because $g^{(3)}$ imposes a distance of closest approach for adjacent polymers.

Performing the Ω and ϕ integrals,

$$\begin{aligned} \langle \mathbf{b}_{1232} \rangle = & \frac{81}{128f_0} 8\pi^2 R_{h1} R_{h2}^2 R_{h3} R_{g2}^4 R_{g3}^2 \int_{2a_0}^{\infty} \frac{dR_1}{R_1^2} \int_{2a_0}^{\infty} \frac{dR_2}{R_2^4} [-\cos \theta + \cos^3 \theta - \frac{6}{5} \cos^5 \theta]_{-1}^1 \\ & \times \text{Tr} \langle \hat{\mathbf{R}}_1 \hat{\mathbf{R}}_1 \rangle \end{aligned} \quad (6.3)$$

Making the change of variables

$$\begin{aligned} R_1 & \rightarrow 2a_0 r, dR_1 \rightarrow 2a_0 dr \\ R_2 & \rightarrow 2a_0 s, dR_2 \rightarrow 2a_0 ds \end{aligned} \quad (6.4)$$

where

$$\begin{aligned} R_1 : 2a_0 \rightarrow \infty & \Rightarrow r : 2 \rightarrow \infty \\ R_2 : 2a_0 \rightarrow \infty & \Rightarrow s : 2 \rightarrow \infty, \end{aligned} \quad (6.5)$$

and evaluating the θ integral leads to

$$\langle \mathbf{b}_{1232} \rangle = \frac{81\pi^2}{16f_0} R_{h1} R_{h2}^2 R_{h3} R_{g2}^4 R_{g3}^2 \int_2^\infty \frac{a_0 dr}{a_0^2 r^2} \int_2^\infty \frac{a_0 ds}{a_0^4 s^4} \left(\frac{12}{5}\right) \text{Tr} \langle \hat{\mathbf{R}}_1 \hat{\mathbf{R}}_1 \rangle \quad (6.6)$$

Performing the r and s integrals and replacing the ensemble average $\langle \mathbf{b}_{1232} \rangle$ in equation 6.1 with its value from equation 6.6,

$$B'_3 = \frac{81\pi^2}{4 \cdot 5} \frac{R_{h1} R_{h2}^2 R_{h3} R_{g2}^4 R_{g3}^2}{a_0^4} \left[-\frac{1}{r}\right]_2^\infty \left[-\frac{1}{3s^3}\right]_2^\infty \text{Tr} \langle \hat{\mathbf{R}}_1 \hat{\mathbf{R}}_1 \rangle \quad (6.7)$$

Since there is no preferred direction for \mathbf{R}_1 , $\text{Tr} \langle \hat{\mathbf{R}}_1 \hat{\mathbf{R}}_1 \rangle = \hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_1 = 1$. Using this result and evaluating the brackets leads to

$$B'_3 = \frac{27\pi^2}{64 \cdot 5} \frac{R_{h1} R_{h2}^2 R_{h3} R_{g2}^4 R_{g3}^2}{a_0^4} \quad (6.8)$$

for B'_3 , the c^2 coefficient due to the figure-eight scattering diagram in the pseudovirial expansion of the self-diffusion coefficient.

This value for B'_3 is much greater than Phillies' B_3 , see equation 3.52, so Phillies' B_3 may be neglected.

6.2 Monte Carlo Routine

A Monte Carlo routine flyind.for was run to perform an ensemble average of \mathbf{b}_{1232} (see appendix C). For 5×10^7 iterations, the output agreed with the analytic calculation

of $\langle \mathbf{b}_{1232} \rangle$ to within 2 %.

Chapter 7

Discussion

B'_3 , the coefficient in the pseudovirial expansion of the self-diffusion coefficient due to trios of polymers interacting hydrodynamically via a figure-eight scattering diagram, a c^2 term, has been determined. The figure-eight diagram applies when polymer one is given an initial translational velocity and produces a wake influencing polymer two; polymer two's subsequent wake influences polymer three; the scattered wake from polymer three acts back on polymer two; and finally, to close the ring, this wake is rescattered, influencing polymer one's motions.

B'_3 resulted from an ensemble average of \mathbf{b}_{1232} , the self-interaction tensor for trios of polymers interacting in a figure-eight scattering diagram. \mathbf{b}_{1232} is derived from \mathbf{T}_{54321} , the cross-interaction tensor for quintets of polymers, by requiring that $\mathbf{R}_4 = -\mathbf{R}_1$ and $\mathbf{R}_3 = -\mathbf{R}_2$. \mathbf{T}_{54321} is determined from $\mathbf{u}^{(4)}$, the wake scattered by the fourth polymer.

The three forces considered in this derivation are the hydrodynamic coupling of

nearby beads, the excluded volume, and the bond-coupling force linking neighboring beads in a polymer. An approximation was made in that the MacLaurin series expansion of the Oseen tensor was taken to $O\left(\frac{s}{r}\right)$ only, giving the longest-range part of the interaction. A higher order expansion would have required more care in the treatment of $g(s)$, which was taken as a function of s only. $g^{(n)}$, the n -polymer distribution functions were taken to be zero for any argument $< 2a_0$, and flat otherwise.

The criterion for self-similarity in this system, that B'_3 , polymer three's effect on the self-diffusion coefficient of polymer one, may be expressed as its effect on polymer two, which, in turn, effects polymer one. This effect of polymer three on polymer two, the $D_s(c)$ on the right-hand-side of equation 1, is found to be the same as the effect of polymer two only on polymer one, the $D_s(c)$ on the left-hand-side of equation 2 as given by the first two terms on the equation's right-hand-side. Mathematically speaking, in equation 2, if a D_0 may be replaced by a $D_s(c)$ to arrive at equation 1, subject to the constraint that $\beta = \alpha^2$, self-similarity is said to have been exhibited.

Equation 6.8 for B'_3 is

$$B'_3 = \frac{27\pi^2}{64 \cdot 5} \frac{R_{h1} R_{h2}^2 R_{h3} R_{g2}^4 R_{g3}^2}{a_0^4}.$$

Equation 3.50 for B_2 is

$$B_2 = -\frac{9}{16} \frac{R_{h1}R_{h2}}{a_0R_g} \left(\frac{4\pi}{3} R_g^3 \right).$$

For self-similarity to hold,

$$B'_3 = B_2^2, \quad (7.1)$$

or,

$$\frac{27\pi^2}{64 \cdot 5} \frac{R_{h1}R_{h2}^2R_{h3}R_{g2}^4R_{g3}^2}{a_0^4} = \left[-\frac{9}{16} \frac{R_{h1}R_{h2}}{a_0R_g} \left(\frac{4\pi}{3} R_g^3 \right) \right]^2. \quad (7.2)$$

Since the polymers are identical, $R_{h1} = R_{h2} = R_{h3} = R_h$ and $R_{g2} = R_{g3} = R_g$.

Therefore, equation 7.2 gives

$$\frac{R_g^2}{a_0^2} = \frac{20}{3} \quad (7.3)$$

or,

$$\frac{a_0}{R_g} = \sqrt{\frac{3}{20}} \approx 0.39 \quad (7.4)$$

as a condition for self-similarity in the self-diffusion coefficient for trios of polymers in solution interacting hydrodynamically via a figure-eight scattering diagram.

Quantitative representative data on polymer chain sizes are found in Adam and Delsanti[10]: for 1.27×10^6 Da polystyrene in benzene, their work shows $R_g = 62.1\text{nm}$, and, with $\eta = 0.6\text{cP}$, $R_h = 38.0\text{nm}$. A Phillies model[11] works for the distance of closest approach $a_0 \approx 0.18R_g$. Substituting into equation 7.3, it is found

that

$$B'_3 \approx 4.6B_2^2. \tag{7.5}$$

A value of $a_0 \approx 0.39R_g$ would give an equality in equation 7.3.

In conclusion, it may be stated that there is substantial support for self-similarity being exhibited the the figure-eight scattering diagram. That is, a D_0 may be replaced by $D_s(c)$ in equation 2 to arrive at equation 1, where $D_s(c)$ is given by the first two terms on the right hand side of equation 2, requiring that $\beta = \alpha^2$.

Appendix A

polyalg.for

```
PROGRAM POLYALG
!THIS PROGRAM TAKES ALGEBRAIC EXPRESSIONS FOR THE FLUID VELOCITIES
!AND POLYMER ROTATIONAL VELOCITIES TO EVALUATE THEM.
!
REAL*8 V(3),R1(3),R2(3),R3(3),R4(3),R1M,R2M,R3M,R4M
REAL*8 CROM2(3),CROM3(3)
REAL*8 R1R2,R1R3,R2R3,R1V
REAL*8 COU2,OM2(3),FACTU2,U2(3),COU3,F11,F12,F2,T2,T3
REAL*8 COOM3,OM3(3),U3(3),OM2M,U2M,OM3M,U3M,U4MAG
REAL*8 COOM4,CROM41(3),CROM42(3),T3OM4,T4OM4,T2OM(3),OM4(3),OM4M
REAL*8 COGRA,R1GRA(3),R2GRA(3),R3GRA(3),GRR(3)
REAL*8 COA,T1A,T2A,T3A,A,R3R4,COU4,T1U4(3),R4GRA,CT2U4
REAL*8 T2U4,T3U4,T4U4,U4(3),T1R3GRA,T2R3GRA,COU4T4,COTXU4
REAL*8 COT4U4A,COT4U4B
REAL*8 POL13,POL14,POL24,POL24A,T5U4,T6U4
!CT2U4=A*R4M*R4M+R4GRA*R3R4
INTEGER I
OPEN(UNIT=8,FILE="polyalg.dat")
OPEN(UNIT=7,FILE="U4alg.dat")
OPEN(UNIT=9,FILE="alginput.dat")
!
!SET THE VALUES FOR V1 AND THE R'S.
!
V(1)=1.5
V(2)=0.0
V(3)=0.0
R1(1)=3.2147
R1(2)=-2.3417
R1(3)=7.1234
```

```

R2(1)=2.3417
R2(2)=7.1342
R2(3)=2.7431
R3(1)=-4.3172
R3(2)=-4.3721
R3(3)=1.7324
R4(1)=-2.3741
R4(2)=4.7123
R4(3)=1.2347
DO 100 I=1,3
    READ(9,*)R1(I)
100 CONTINUE
DO 110 I=1,3
    READ(9,*)R2(I)
110 CONTINUE
DO 120 I=1,3
    READ(9,*)R3(I)
120 CONTINUE
DO 130 I=1,3
    READ(9,*)R4(I)
130 CONTINUE
CLOSE(9,STATUS='KEEP')
!DO 98 I=1,3
!   R3(I)=(-1.0)*R2(I)
!   R4(I)=(-1.0)*R1(I)
98 CONTINUE
!
R1M=SQRT(R1(1)*R1(1)+R1(2)*R1(2)+R1(3)*R1(3))
R2M=SQRT(R2(1)*R2(1)+R2(2)*R2(2)+R2(3)*R2(3))
R3M=SQRT(R3(1)*R3(1)+R3(2)*R3(2)+R3(3)*R3(3))
R4M=SQRT(R4(1)*R4(1)+R4(2)*R4(2)+R4(3)*R4(3))
V1M=SQRT(V(1)*V(1)+V(2)*V(2)+V(3)*V(3))
!
!COMPUTE THE DOT PRODUCTS OF THE UNIT VECTORS(EXCEPTING V)
!
R1V=(R1(1)*V(1)+R1(2)*V(2)+R1(3)*V(3))/R1M
R1R2=(R1(1)*R2(1)+R1(2)*R2(2)+R1(3)*R2(3))/R1M/R2M
R1R3=(R1(1)*R3(1)+R1(2)*R3(2)+R1(3)*R3(3))/R1M/R3M
R2R3=(R2(1)*R3(1)+R2(2)*R3(2)+R2(3)*R3(3))/R2M/R3M
R1R4=(R1(1)*R4(1)+R1(2)*R4(2)+R1(3)*R4(3))/R1M/R4M
R2R4=(R2(1)*R4(1)+R2(2)*R4(2)+R2(3)*R4(3))/R2M/R4M
R3R4=(R3(1)*R4(1)+R3(2)*R4(2)+R3(3)*R4(3))/R3M/R4M
!
```

```

!COMPUTE OMEGA2, FIRST FINDING ITS CROSS-PRODUCT FACTOR.
!
CROM2(1)=(R1(2)*V(3)-R1(3)*V(2))/R1M
CROM2(2)=(R1(3)*V(1)-R1(1)*V(3))/R1M
CROM2(3)=(R1(1)*V(2)-R1(2)*V(1))/R1M
DO 10 I=1,3
    OM2(I)=(-0.75)/(R1M*R1M)*CROM2(I)
10 CONTINUE
!
!COMPUTE U2.
!
COU2=(-9.0)/16.0/(R1M*R1M)/(R2M*R2M*R2M)
FACTU2=(1.0-3.0*R1R2*R1R2)*R1V
DO 20 I=1,3
    U2(I)=COU2*FACTU2*R2(I)
20 CONTINUE
!
!COMPUTE OMEGA3.
!
CROM3(1)=(R1(2)*R2(3)-R1(3)*R2(2))/R1M/R2M
CROM3(2)=(R1(3)*R2(1)-R1(1)*R2(3))/R1M/R2M
CROM3(3)=(R1(1)*R2(2)-R1(2)*R2(1))/R1M/R2M
COOM3=27.0/16.0/(R1M*R1M)/(R2M*R2M*R2M)
DO 30 I=1,3
    OM3(I)=COOM3*R1V*R1R2*CROM3(I)
30 CONTINUE
!
!COMPUTE U3.
!
COU3=27.0/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M)
F11=1.0-3.0*R1R2*R1R2
F12=1.0-3.0*R2R3*R2R3
F2=6.0*R1R2*R2R3
T2=F2*R1R3
T3=F2*R1R2*R2R3
DO 40 I=1,3
    U3(I)=COU3*(F11*F12-(T2-T3))*R1V*R3(I)/R3M
40 CONTINUE
!
!COMPUTE MAGNITUDES.
!
OM2M=SQRT(OM2(1)*OM2(1)+OM2(2)*OM2(2)+OM2(3)*OM2(3))
U2M=SQRT(U2(1)*U2(1)+U2(2)*U2(2)+U2(3)*U2(3))

```

```

OM3M=SQRT(OM3(1)*OM3(1)+OM3(2)*OM3(2)+OM3(3)*OM3(3))
U3M=SQRT(U3(1)*U3(1)+U3(2)*U3(2)+U3(3)*U3(3))
!
!COMPUTE OMEGA4
!
COOM4=(-81.0)/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)*R1V
CROM41(1)=(R1(2)*R3(3)-R1(3)*R3(2))/R1M/R3M
CROM41(2)=(R1(3)*R3(1)-R1(1)*R3(3))/R1M/R3M
CROM41(3)=(R1(1)*R3(2)-R1(2)*R3(1))/R1M/R3M
CROM42(1)=(R2(2)*R3(3)-R2(3)*R3(2))/R2M/R3M
CROM42(2)=(R2(3)*R3(1)-R2(1)*R3(3))/R2M/R3M
CROM42(3)=(R2(1)*R3(2)-R2(2)*R3(1))/R2M/R3M
T3OM4=R1R2*R2R3
T4OM4=R1R2*R1R2*R2R3
DO 50 I=1,3
    T2OM(I)=CROM42(I)*(R2R3-T4OM4+R1R2*R1R3)
50 CONTINUE
!
DO 60 I=1,3
    OM4(I)=COOM4*(CROM41(I)*R1R2*R2R3+T2OM(I))
60 CONTINUE
!
OM4M=SQRT(OM4(1)*OM4(1)+OM4(2)*OM4(2)+OM4(3)*OM4(3))
!
!COMPUTE VARIOUS PIECES OF TERMS IN U4.
!GRAD A WRT R3
!
COGRA=(-81.0)/32.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M)**2.0
DO 70 I=1,3
    R1GRA(I)=R1R2*R2R3*R1(I)
    R2GRA(I)=((1.0-5.0*R1R2*R1R2)*R2R3+R1R2*R1R3)*R2(I)
    T1R3GRA=1.0-3.0*R1R2*R1R2-10.0*R1R2*R1R3*R2R3
    T2R3GRA=(-5.0)*(1.0-5.0*R1R2*R1R2)*R2R3*R2R3
    !R3GRA(I)=((1.0-3.0*R1R2*R1R2)-10.0*R1R2*R1R3*R2R3-5.0*(
c        (1.0-5.0*R1R2*R1R2)*R2R3*R2R3)*R3(I)/2.0
    R3GRA(I)=(T1R3GRA+T2R3GRA)*R3(I)/2.0
    GRRR(I)=R1GRA(I)+R2GRA(I)+R3GRA(I)
70 CONTINUE
!
!OUTPUT GRAD A WRT R
!
!COMPUTE A
!
```

```

COA=27.0/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)
T1A=(1.0-3.0*R1R2*R1R2)*(1.0-3.0*R2R3*R2R3)
T2A=6.0*R1R2*R1R3*R2R3
T3A=6.0*R1R2*R1R2*R2R3*R2R3
A=COA*(T1A-T2A+T3A)
!
!OUTPUT A
!
!COMPUTE THE FOUR TERMS IN U4
!
!COU4=1.0/4.0/(R4M*R4M*R4M)
!T1U4(1)=2.0*(OM4(2)*R4(3)-OM4(3)*R4(2))*R1V
!T1U4(2)=2.0*(OM4(3)*R4(1)-OM4(1)*R4(3))*R1V
!T1U4(3)=2.0*(OM4(1)*R4(2)-OM4(2)*R4(1))*R1V
!
!R4GRA=R4(1)*GRRRA(1)+R4(2)*GRRRA(2)+R4(3)*GRRRA(3)
!CT2U4=R4(1)*(A*R4(1)+R4GRA*R3(1))+R4(2)*(A*R4(2)+R4GRA*R3(2))
!R4(3)*(A*R4(3)+R4GRA*R3(3))
!CT2U4=A*R4M*R4M+R4GRA*R3R4
!DO 80 I=1,3
!   T2U4(I)=3.0*R4(I)*CT2U4/(R4M*R4M)*R1V
!   T3U4(I)=A*R4(I)+GRRRA(I)*R3R4*R1V
!   T4U4(I)=A*R4(I)+R4GRA*R3(I)*R1V
!   U4(I)=COU4*(T1U4(I)-T2U4(I)+T3U4(I)-T4U4(I))
!CONTINUE
!
!FINALLY, MAYBE I HAVE AN EXPRSSION FOR U4
!
COU4=(-3.0)/4.0/(R4M*R4M)*R1V
COTXU4=(-81.0)/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)*R3R4
!T2U4=R1R2*R2R3*R1R4
!T3U4=(R2R3*(1.0-5.0*R1R2*R1R2)+R1R2*R1R3)*R2R4
!COT4U4A=1.0-3.0*R1R2*R1R2-10.0*R1R2*R2R3*R1R3
!COT4U4B=5.0*R2R3*R2R3*(1.0-5.0*R1R2*R1R2)
!COT4U4=COT4U4A-COT4U4B
!T4U4=R3R4/2.0*COT4U4
POL13=R1R3-R1R2*R2R3
POL14=R1R4-R1R2*R2R4
POL24=R2R4-R2R3*R3R4
POL24A=R2R4-5.0*R2R3*R3R4
T2U4=T1A*R3R4
T3U4=2.0*R1R2*POL13*POL24A
T4U4=2.0*(1.0-3.0*R1R2*R1R2)*R2R3*POL24

```

```

      T5U4=2.0*R1R2*R2R3*POL14
      DO 80 I=1,3
          U4(I)=COU4*R4(I)/R4M*(A+COTXU4*(T2U4+T3U4+T4U4+T5U4))
80      CONTINUE
      U4MAG = SQRT((U4(1)*U4(1)+U4(2)*U4(2)+U4(3)*U4(3)))
      !
      ! I/O
      !WRITE(7,97)T1U4
      !WRITE(7,97)T2U4
      !WRITE(7,97)T3U4
      !WRITE(7,97)T4U4
      WRITE(7,97)U4
      !
      !
      WRITE(8,45)
      WRITE(8,21)
      WRITE(8,18)V
      WRITE(8,19)V1M
      WRITE(8,21)
      WRITE(8,11)R1
      WRITE(8,22)R1M
      WRITE(8,21)
      WRITE(8,13)OM2
      WRITE(8,24)OM2M
      WRITE(8,21)
      WRITE(8,12)R2
      WRITE(8,23)R2M
      WRITE(8,21)
      WRITE(8,14)U2
      WRITE(8,25)U2M
      WRITE(8,21)
      WRITE(8,16)OM3
      WRITE(8,27)OM3M
      WRITE(8,21)
      WRITE(8,15)R3
      WRITE(8,26)R3M
      WRITE(8,21)
      WRITE(8,17)U3
      WRITE(8,28)U3M
      WRITE(8,21)
      WRITE(8,49)R4
      WRITE(8,51)R4M
      WRITE(8,21)

```



```

WRITE(8,47)OM4
WRITE(8,48)OM4M
WRITE(8,52)U4
WRITE(8,*)U4MAG
97  FORMAT(3E14.7)
11  FORMAT("      R1x=",E14.7,"      R1y=",E14.7,"      R1z=",E14.7)
12  FORMAT("      R2x=",E14.7,"      R2y=",E14.7,"      R2z=",E14.7)
13  FORMAT("  Omega2x=",E14.7,"  Omega2y=",E14.7,"  Omega2z=",E14.7)
14  FORMAT("      U2x=",E14.7,"      U2y=",E14.7,"      U2z=",E14.7)
15  FORMAT("      R3x=",E14.7,"      R3y=",E14.7,"      R3z=",E14.7)
16  FORMAT("  Omega3x=",E14.7,"  Omega3y=",E14.7,"  Omega3z=",E14.7)
17  FORMAT("      U3x=",E14.7,"      U3y=",E14.7,"      U3z=",E14.7)
18  FORMAT("      V1x=",E14.7,"      V1y=",E14.7,"      V1z=",E14.7)
49  FORMAT("      R4x=",E14.7,"      R4y=",E14.7,"      R4z=",E14.7)
51  FORMAT("      |R4|=",E14.7)
47  FORMAT("  Omega4x=",E14.7,"  Omega4y=",E14.7,"  Omega4z=",E14.7)
48  FORMAT(" |Omega4|=",E14.7)
19  FORMAT("      |V1|=",E14.7)
21  FORMAT("")
22  FORMAT("      |R1|=",E14.7)
23  FORMAT("      |R2|=",E14.7)
24  FORMAT(" |Omega2|=",E14.7)
25  FORMAT("      |U2|=",E14.7)
26  FORMAT("      |R3|=",E14.7)
27  FORMAT(" |Omega3|=",E14.7)
28  FORMAT("      |U3|=",E14.7)
45  FORMAT("These are the algebraic results.")
52  FORMAT(3E14.7)
      CLOSE(7)
      CLOSE(8)
      STOP
      END

```

Appendix B

hydrpoly.for

```
PROGRAM HYDRPOLY
REAL*8 V(3),R(3),U_1(3),RP(3,3),RM(3,3),A
INTEGER I,J
REAL*8 RPX(3),RPY(3),RPZ(3),RMX(3),RMY(3),RMZ(3)
REAL*8 UPX(3),UPY(3),UPZ(3),UMX(3),UMY(3),UMZ(3)
REAL*8 UPLSA(3,3),UMINA(3,3),OM(3),R2(3),UN(3)
REAL*8 UNPX(3),UNPY(3),UNPZ(3),UNMX(3),UNMY(3),UNMZ(3),OM3(3)
REAL*8 R3(3),U3(3),OM4(3),R4(3),U4(3)
REAL*8 V1MAG,R1MAG,U1MAG,OM2MAG,R2MAG,U2MAG,OM3MAG,R3MAG,U3MAG
REAL*8 OM4MAG,R4MAG,U4MAG,OM2C(3),OM3C(3),OM4C(3)
REAL*8 U2C(3),U2CM,U3C(3),U3CM,U4C(3),U4CM,OM2CM,OM3CM,OM4CM
REAL*8 FPX(3),FPY(3),FPZ(3),FMX(3),FMY(3),FMZ(3)
REAL*8 FPLSA(3,3),FMINA(3,3),TEST(3)
OPEN(UNIT=4,FILE="testcurl.dat")
OPEN(UNIT=8,FILE="hydrpoly.dat")
OPEN(UNIT=9,FILE="hydrpol.dat")
OPEN(UNIT=10,FILE="alginput.dat")
V(1)=1.5
V(2)=0.0
V(3)=0.0
V1MAG=SQRT(V(1)*V(1)+V(2)*V(2)+V(3)*V(3))
R(1)=3.2147
R(2)=-2.3417
R(3)=7.1234
DO 201 I=1,3
    READ(10,*)R(I)
201 CONTINUE
R1MAG=SQRT(R(1)*R(1)+R(2)*R(2)+R(3)*R(3))
CALL U1(V,R,U_1)
U1MAG=SQRT(U_1(1)**2.+U_1(2)**2.+U_1(3)**2.)
```

```

!
!NOTE WELL, THE "FIRST" POLYMER IS POLYMER ZERO.
!
!INITIALIZE RPLUS, RP(3,3), AND RMINUS, RM(3,3).  THESE ARE THE VALUES
!OF R WHICH WILL BE USED TO COMPUTE THE VALUES OF U'S FOR TAKING ITS
!DERIVATIVES, BOTH THE GRADIENT AND THE CURL.  I WILL USE AN INCREMENT,
!A, ADDED TO, AND SUBTRACTED FROM, EACH OF THE THREE COORDINATES.
!THE DIVISOR WILL THEN, NECESSARILY BE 2*A.
!
DO 10 I=1,3
    DO 20 J=1,3
        RP(I,J)=R(J)
        RM(I,J)=R(J)
20    CONTINUE
10    CONTINUE
A=0.005
DO 30 I=1,3
    RP(I,I)=R(I)+A
    RM(I,I)=R(I)-A
30    CONTINUE
!
!I WILL PUT THE VALUES OF THE VARIOUS R'S INTO VECTORS IN ORDER TO
!USE THE ROUTINE U1.
!
DO 40 I=1,3
    RPX(I)=RP(1,I)
    RPY(I)=RP(2,I)
    RPZ(I)=RP(3,I)
    RMX(I)=RM(1,I)
    RMY(I)=RM(2,I)
    RMZ(I)=RM(3,I)
40    CONTINUE
!
!
!I WILL CALL THE ROUTINE, U1, FOR THESE SIX VALUES OF R, AND THEN USE
!THE RESULTING VALUES OF U1 TO COMPUTE ITS CURL, GIVING OMEGA2, THE
!ROTATIONAL VELOCITY OF POLYMER TWO.
!
!
CALL U1(V,RPX,UPX)
CALL U1(V,RPY,UPY)
CALL U1(V,RPZ,UPZ)
CALL U1(V,RMX,UMX)

```

```

CALL U1(V,RMY,UMY)
CALL U1(V,RMZ,UMZ)
!
!I WILL PUT THE VALUES OF U1 INTO ONE OF TWO MATRICES, UPLSA(3,3),
!OR UMINA(3,3). UPLSA HOLDS THE VALUES OF U1 FOR A ADDED TO X,Y OR Z.
!UMINA HOLDS THE VALUES OF U1 FOR A SUBTRACTED FROM X, Y OR Z.
!
DO 50 I=1,3
    UPLSA(I,1)=UPX(I)
    UPLSA(I,2)=UPY(I)
    UPLSA(I,3)=UPZ(I)
    UMINA(I,1)=UMX(I)
    UMINA(I,2)=UMY(I)
    UMINA(I,3)=UMZ(I)
50 CONTINUE
!
!I WILL CALL THE SUBROUTINE CURL WITH THESE VALUES OF U1, TO COMPUTE
!ITS CURL, HALF OF WHICH IS OMEGA2, THE ROTATIONAL VELOCITY OF
!POLYMER TWO.
!
CALL CURL(UPLSA,UMINA,A,OM)
!
OM2MAG=SQRT(OM(1)*OM(1)+OM(2)*OM(2)+OM(3)*OM(3))
!
!WE NOW HAVE U1(R) AND OM(R). NEED TO WRITE A ROUTINE FOR UN+1,
!N>=1. UN+1 NEEDS OMN+1 AT THE SAME POINT. CAN JUST USE THE
!U'S NEEDED TO DETERMINE ONEGA.
!FIRST USE SUBROUTINE TO CALCULATE U(R).
!THEN USE SUBROUTINE TOO CALCULATE U(R+/-A).
!GIVE THE SUBROUTINE RNP1(3), OM(3), UPLSA(3,3),
!AND UMINA(3,3). RETURN UNPLS1(3).
!
R2(1)=2.3417
R2(2)=7.1342
R2(3)=2.7431
DO 210 I=1,3
    READ(10,*)R2(I)
210 CONTINUE
R2MAG=SQRT(R2(1)*R2(1)+R2(2)*R2(2)+R2(3)*R2(3))
!
!COMPUTE U2
!
!N IS THE GENERATION.

```

```

!
N=2
!
CALL UNPLS1(R2,N,OM,UPLSA,UMINA,A,UN)
!
U2MAG=SQRT(UN(1)*UN(1)+UN(2)*UN(2)+UN(3)*UN(3))
!
!COMPUTE OMEGA3.
!
DO 70 I=1,3
    DO 80 J=1,3
        RP(I,J)=R2(J)
        RM(I,J)=R2(J)
80    CONTINUE
70    CONTINUE
DO 90 I=1,3
    RP(I,I)=RP(I,I)+A
    RM(I,I)=RM(I,I)-A
90    CONTINUE
!
DO 100 I=1,3
    RPX(I)=RP(1,I)
    RPY(I)=RP(2,I)
    RPZ(I)=RP(3,I)
    RMX(I)=RM(1,I)
    RMY(I)=RM(2,I)
    RMZ(I)=RM(3,I)
100    CONTINUE
!
CALL UNPLS1(RPX,N,OM,UPLSA,UMINA,A,UNPX)
CALL UNPLS1(RPY,N,OM,UPLSA,UMINA,A,UNPY)
CALL UNPLS1(RPZ,N,OM,UPLSA,UMINA,A,UNPZ)
CALL UNPLS1(RMX,N,OM,UPLSA,UMINA,A,UNMX)
CALL UNPLS1(RMY,N,OM,UPLSA,UMINA,A,UNMY)
CALL UNPLS1(RMZ,N,OM,UPLSA,UMINA,A,UNMZ)
!
DO 110 I=1,3
    UPLSA(I,1)=UNPX(I)
    UPLSA(I,2)=UNPY(I)
    UPLSA(I,3)=UNPZ(I)
    UMINA(I,1)=UNMX(I)
    UMINA(I,2)=UNMY(I)
    UMINA(I,3)=UNMZ(I)

```

```

110 CONTINUE
!
!CALL CURL TO COMPUTE OMEGA3
!
CALL CURL(UPLSA,UMINA,A,OM3)
!
OM3MAG=SQRT(OM3(1)*OM3(1)+OM3(2)*OM3(2)+OM3(3)*OM3(3))
!
!CALL UNPLS1 TO COMPUTE U3
!
R3(1)=-4.3172
R3(2)=-4.3721
R3(3)=1.7324
DO 220 I=1,3
    READ(10,*)R3(I)
220 CONTINUE
!R3(1)=(-1.0)*R2(1)
!R3(2)=(-1.0)*R2(2)
!R3(3)=(-1.0)*R2(3)
R3MAG=SQRT(R3(1)*R3(1)+R3(2)*R3(2)+R3(3)*R3(3))
!
!OM3(1)=-0.000004090333
!OM3(2)=0.0000005618257
!OM3(3)=0.000002030606
!WRITE(9,85)
85 FORMAT("Omega3 set equal to the algebraic results.")
!
OM2C(1)=0.0
OM2C(2)=-0.01475751
OM2C(3)=-0.004851289
OM3C(1)=-0.000004090333
OM3C(2)=0.0000005618257
OM3C(3)=0.000002030606
OM4C(1)=0.000000006005069
OM4C(2)=-0.000000008248233
OM4C(3)=-0.000000002981159
!
OM2CM=SQRT(OM2C(1)*OM2C(1)+OM2C(2)*OM2C(2)+OM2C(3)*OM2C(3))
OM3CM=SQRT(OM3C(1)*OM3C(1)+OM3C(2)*OM3C(2)+OM3C(3)*OM3C(3))
OM4CM=SQRT(OM4C(1)*OM4C(1)+OM4C(2)*OM4C(2)+OM4C(3)*OM4C(3))
!
N=3
!

```

```

CALL UNPLS1(R3,N,OM3,UPLSA,UMINA,A,U3)
U3MAG=SQRT(U3(1)*U3(1)+U3(2)*U3(2)+U3(3)*U3(3))
!
!PREPARE TO COMPUTE THE GRAD TERMS IN OMEGA4 WITH UPLSA AND UMINA.
!
DO 120 I=1,3
    DO 130 J=1,3
        RP(I,J)=R3(J)
        RM(I,J)=R3(J)
130    CONTINUE
120    CONTINUE
DO 140 I=1,3
    RP(I,I)=RP(I,I)+A
    RM(I,I)=RM(I,I)-A
140    CONTINUE
!
DO 150 I=1,3
    RPX(I)=RP(1,I)
    RPY(I)=RP(2,I)
    RPZ(I)=RP(3,I)
    RMX(I)=RM(1,I)
    RMY(I)=RM(2,I)
    RMZ(I)=RM(3,I)
150    CONTINUE
!
!TESTING CURL SUBROUTINE WITH Y XHAT - X YHAT, or  $f(x,y,z)=(y,-x,0)$ 
!
CALL COMPUTE(RPX,A,FPX)
CALL COMPUTE(RPY,A,FPY)
CALL COMPUTE(RPZ,A,FPZ)
CALL COMPUTE(RMX,A,FMX)
CALL COMPUTE(RMY,A,FMY)
CALL COMPUTE(RMZ,A,FMZ)
DO 200 I=1,3
    FPLSA(I,1)=FPX(I)
    FPLSA(I,2)=FPY(I)
    FPLSA(I,3)=FPZ(I)
    FMINA(I,1)=FMX(I)
    FMINA(I,2)=FMY(I)
    FMINA(I,3)=FMZ(I)
200    CONTINUE
CALL CURL(FPLSA,FMINA,A,TEST)
!

```

```

WRITE(4,99)TEST
99  FORMAT(3F14.7)
    CALL UNPLS1(RPX,N,OM3,UPLSA,UMINA,A,UNPX)
    CALL UNPLS1(RPY,N,OM3,UPLSA,UMINA,A,UNPY)
    CALL UNPLS1(RPZ,N,OM3,UPLSA,UMINA,A,UNPZ)
    CALL UNPLS1(RMX,N,OM3,UPLSA,UMINA,A,UNMX)
    CALL UNPLS1(RMY,N,OM3,UPLSA,UMINA,A,UNMY)
    CALL UNPLS1(RMZ,N,OM3,UPLSA,UMINA,A,UNMZ)
    !
    DO 160 I=1,3
        UPLSA(I,1)=UNPX(I)
        UPLSA(I,2)=UNPY(I)
        UPLSA(I,3)=UNPZ(I)
        UMINA(I,1)=UNMX(I)
        UMINA(I,2)=UNMY(I)
        UMINA(I,3)=UNMZ(I)
160  CONTINUE
    !
    !CALL CURL TO COMPUTE OMEGA4
    !
    CALL CURL(UPLSA,UMINA,A,OM4)
    !
    OM4MAG=SQRT(OM4(1)*OM4(1)+OM4(2)*OM4(2)+OM4(3)*OM4(3))
    !
    !CALL UNPLS1 TO COMPUTE U4
    !
    R4(1)=(-2.3741)
    R4(2)=4.7123
    R4(3)=1.2347
    DO 230 I=1,3
        READ(10,*)R4(I)
230  CONTINUE
    CLOSE(10,STATUS='KEEP')
    !R4(1)=(-1.0)*R(1)
    !R4(2)=(-1.0)*R(2)
    !R4(3)=(-1.0)*R(3)
    R4MAG=SQRT(R4(1)*R4(1)+R4(2)*R4(2)+R4(3)*R4(3))
    !
    N=4
    CALL UNPLS1(R4,N,OM4,UPLSA,UMINA,A,U4)
    U4MAG=SQRT(U4(1)*U4(1)+U4(2)*U4(2)+U4(3)*U4(3))
    !
    U2C(1)=-0.00002116203

```



```

U2C(2)=-0.00006447204
U2C(3)=-0.00002478950
U3C(1)=0.00000006213650
U3C(2)=0.0000001893044
U3C(3)=0.00000007278756
U4C(1)=-0.0000000001182996
U4C(2)=0.00000000008617355
U4C(3)=-0.0000000002621381
U2CM=SQRT(U2C(1)*U2C(1)+U2C(2)*U2C(2)+U2C(3)*U2C(3))
U3CM=SQRT(U3C(1)*U3C(1)+U3C(2)*U3C(2)+U3C(3)*U3C(3))
U4CM=SQRT(U4C(1)*U4C(1)+U4C(2)*U4C(2)+U4C(3)*U4C(3))
!OUTPUT VARIOUS RESULTS
!
WRITE(8,45)
WRITE(8,21)
WRITE(8,46)A
WRITE(8,21)
WRITE(8,11)V
WRITE(8,31)V1MAG
WRITE(8,21)
WRITE(8,15)R
WRITE(8,32)R1MAG
WRITE(8,21)
WRITE(8,16)U_1
WRITE(8,33)U1MAG
WRITE(8,21)
WRITE(8,17)OM
WRITE(8,34)OM2MAG
WRITE(8,21)
WRITE(8,18)R2
WRITE(8,36)R2MAG
WRITE(8,21)
WRITE(8,19)UN
WRITE(8,37)U2MAG
WRITE(8,21)
WRITE(8,22)OM3
WRITE(8,38)OM3MAG
WRITE(8,21)
WRITE(8,28)R3
WRITE(8,39)R3MAG
WRITE(8,21)
WRITE(8,23)U3
WRITE(8,41)U3MAG

```

```
WRITE(8,21)
WRITE(8,24)OM4
WRITE(8,42)OM4MAG
WRITE(8,21)
WRITE(8,26)R4
WRITE(8,43)R4MAG
WRITE(8,21)
WRITE(8,27)U4
WRITE(8,44)U4MAG
WRITE(9,21)
WRITE(9,74)
WRITE(9,21)
WRITE(9,75)
WRITE(9,21)
WRITE(9,79)A
WRITE(9,21)
WRITE(9,76)
WRITE(9,21)
WRITE(9,77)
WRITE(9,21)
WRITE(9,78)V,V1MAG
WRITE(9,21)
WRITE(9,21)
WRITE(9,51)
WRITE(9,21)
WRITE(9,52)R,R1MAG
WRITE(9,21)
WRITE(9,53)R2,R2MAG
WRITE(9,21)
WRITE(9,54)R3,R3MAG
WRITE(9,21)
WRITE(9,55)R4,R4MAG
WRITE(9,21)
WRITE(9,21)
WRITE(9,56)
WRITE(9,21)
WRITE(9,57)U_1,U1MAG
WRITE(9,21)
WRITE(9,58)UN,U2MAG
WRITE(9,71)U2C,U2CM
WRITE(9,21)
WRITE(9,59)U3,U3MAG
WRITE(9,72)U3C,U3CM
```

```

WRITE(9,21)
WRITE(9,61)U4,U4MAG
WRITE(9,73)U4C,U4CM
WRITE(9,21)
WRITE(9,21)
WRITE(9,62)
WRITE(9,21)
WRITE(9,63)OM,OM2MAG
WRITE(9,67)OM2C,OM2CM
WRITE(9,21)
WRITE(9,64)OM3,OM3MAG
WRITE(9,68)OM3C,OM3CM
WRITE(9,21)
WRITE(9,65)OM4,OM4MAG
WRITE(9,69)OM4C,OM4CM
!WRITE(9,66)
11  FORMAT("      V1x =",E14.7,"      V1y =",E14.7,"      V1z =",E14.7)
15  FORMAT("      R1x =",E14.7,"      R1y =",E14.7,"      R1z =",E14.7)
16  FORMAT("      U1x =",E14.7,"      U1y =",E14.7,"      U1z =",E14.7)
17  FORMAT(" Omega2x =",E14.7," Omega2y =",E14.7," Omega2z =",E14.7)
18  FORMAT("      R2x =",E14.7,"      R2y =",E14.7,"      R2z =",E14.7)
19  FORMAT("      U2x =",E14.7,"      U2y =",E14.7,"      U2z =",E14.7)
21  FORMAT("")
22  FORMAT(" Omega3x =",E14.7," Omega3y =",E14.7," Omega3z =",E14.7)
28  FORMAT("      R3x =",E14.7,"      R3y =",E14.7,"      R3z =",E14.7)
23  FORMAT("      U3x =",E14.7,"      U3y =",E14.7,"      U3z =",E14.7)
24  FORMAT(" Omega4x =",E14.7," Omega4y =",E14.7," Omega4z =",E14.7)
26  FORMAT("      R4x =",E14.7,"      R4y =",E14.7,"      R4z =",E14.7)
27  FORMAT("      U4x =",E14.7,"      U4y =",E14.7,"      U4z =",E14.7)
31  FORMAT("      |V1|=",E14.7)
32  FORMAT("      |R1|=",E14.7)
33  FORMAT("      |U1|=",E14.7)
34  FORMAT(" |Omega2|=",E14.7)
36  FORMAT("      |R2|=",E14.7)
37  FORMAT("      |U2|=",E14.7)
38  FORMAT(" |Omega3|=",E14.7)
39  FORMAT("      |R3|=",E14.7)
41  FORMAT("      |U3|=",E14.7)
42  FORMAT(" |Omega4|=",E14.7)
43  FORMAT("      |R4|=",E14.7)
44  FORMAT("      |U4|=",E14.7)
45  FORMAT("These are the results from the calculus equations.")
46  FORMAT("      The increment for differentiation, A, equals ",E14.7)

```

```

51  FORMAT("Field Point      Rx          Ry          Rz          |R|")
52  FORMAT(" 1  ",3E16.8,E17.8)
53  FORMAT(" 2  ",3E16.8,E17.8)
54  FORMAT(" 3  ",3E16.8,E17.8)
55  FORMAT(" 4  ",3E16.8,E17.8)
56  FORMAT("Field Point      Ux          Uy          Uz          |U|")
57  FORMAT(" 1  ",4E16.8)
58  FORMAT(" 2  ",4E16.8)
59  FORMAT(" 3  ",4E16.8)
61  FORMAT(" 4  ",4E16.8)
62  FORMAT("Polymer      Omegax      Omegay      Omegaz      |Omega|")
63  FORMAT(" 2  ",4E16.8)
64  FORMAT(" 3  ",4E16.8)
65  FORMAT(" 4  ",4E16.8)
66  FORMAT("Polymer      Omegax(Set)  Omegay(Set)  Omegaz(Set)")
67  FORMAT("2,alg",4E16.8)
68  FORMAT("3,alg",4E16.8)
69  FORMAT("4,alg",4E16.8)
71  FORMAT("2,alg",4E16.8)
72  FORMAT("3,alg",4E16.8)
73  FORMAT("4,alg",4E16.8)
74  FORMAT("Fluid motion, Ui, at a field point, Ri, for polymer i+1,")
75  FORMAT("with angular velocity Omega i+1.")
76  FORMAT("Translational velocity of polymer 1, V1.")
77  FORMAT("      Vx          Vy          Vz          |V|")
78  FORMAT(4E16.8)
79  FORMAT("The differential element of integration is equal ",E16.8)
      CLOSE(8)
      CLOSE(9)
      STOP
      END
      SUBROUTINE COMPUTE(R,A,F)
      DIMENSION R(3),F(3)
      REAL*8 R,A,F
      F(1)=R(2)
      F(2)=-1.0*R(1)
      F(3)=0.0
      RETURN
      END
      SUBROUTINE U1(V,R,U_1)
      REAL*8 V(3), R(3), RMAG, COEFF, TERM1, FACT
      REAL*8 U_1(3)
      INTEGER I,J

```

```

!
!THE EQUATION EMPLOYED CAN BE FOUND IN PHILLIES,
!MACROMOLECULES 31 (98) 2317-2327.
!
COEFF=0.75
RMAG=SQRT(R(1)*R(1)+R(2)*R(2)+R(3)*R(3))
FACT=(R(1)*V(1)+R(2)*V(2)+R(3)*V(3))/(RMAG*RMAG)
DO 10 I=1,3
    U_1(I)=COEFF*(V(I)+R(I)*FACT)/RMAG
10    CONTINUE
RETURN
END
SUBROUTINE CURL(UPLSA,UMINA,A,OM)
DIMENSION UPLSA(3,3),UMINA(3,3),OM(3)
REAL*8 A,UPLSA,UMINA,OM
OM(1)=(UPLSA(3,2)-UMINA(3,2)-UPLSA(2,3)+UMINA(2,3))/A/4.
OM(2)=(UPLSA(1,3)-UMINA(1,3)-UPLSA(3,1)+UMINA(3,1))/A/4.
OM(3)=(UPLSA(2,1)-UMINA(2,1)-UPLSA(1,2)+UMINA(1,2))/A/4.
RETURN
END
SUBROUTINE UNPLS1(R,N,OM,UPLSA,UMINA,A,U)
DIMENSION R(3),OM(3),UPLSA(3,3),UMINA(3,3),U(3),CR(3)
DIMENSION GR1(3),GR2(3),GR3(3)
REAL*8 R,OM,UPLSA,UMINA,A,U
REAL*8 CR,Co, RMAG
REAL*8 PUXX,PUXY,PUXZ,PUYX,PUYY,PUYZ,PUZX,PUZY,PUZZ
REAL*8 COR1,COR2,COR3,GR1,GR2,GR3
INTEGER I,J
OPEN(UNIT=7,FILE="U2.dat")
OPEN(UNIT=6,FILE="U3.dat")
OPEN(UNIT=5,FILE="U4.dat")
RMAG=SQRT(R(1)*R(1)+R(2)*R(2)+R(3)*R(3))
!COMPUTE PREFACTOR
Co=1.0/(4.0*RMAG**3.)
!
!COMPUTE CROSS-PRODUCT, CR(I)
!
CR(1)=2.*(OM(2)*R(3)-OM(3)*R(2))
CR(2)=2.*(OM(3)*R(1)-OM(1)*R(3))
CR(3)=2.*(OM(1)*R(2)-OM(2)*R(1))
!
!COMPUTE PARTIALS, PUXX - PARTIAL UX, PARTIAL X; PUXY - PARTIAL UX,
!PARTIAL Y; ETC.

```

```

!
PUXX=(UPLSA(1,1)-UMINA(1,1))/(2.*A)
PUXY=(UPLSA(1,2)-UMINA(1,2))/(2.*A)
PUXZ=(UPLSA(1,3)-UMINA(1,3))/(2.*A)
PUYX=(UPLSA(2,1)-UMINA(2,1))/(2.*A)
PUYY=(UPLSA(2,2)-UMINA(2,2))/(2.*A)
PUYZ=(UPLSA(2,3)-UMINA(2,3))/(2.*A)
PUZX=(UPLSA(3,1)-UMINA(3,1))/(2.*A)
PUZY=(UPLSA(3,2)-UMINA(3,2))/(2.*A)
PUZZ=(UPLSA(3,3)-UMINA(3,3))/(2.*A)
!
!COMPUTE FIRST GRAD TERM, GR1(3)
!FIND COEFFICIENT OF R, COR
!
COR1=R(1)*(R(1)*PUXX+R(2)*PUXY+R(3)*PUXZ)
COR2=R(2)*(R(1)*PUYX+R(2)*PUYY+R(3)*PUYZ)
COR3=R(3)*(R(1)*PUZX+R(2)*PUZY+R(3)*PUZZ)
COR=3.0*(COR1+COR2+COR3)*(-1.0)/(RMAG*RMAG)
GR1(1)=R(1)*COR
GR1(2)=R(2)*COR
GR1(3)=R(3)*COR
!
!COMPUTE SECOND GRAD TERM, GR2(3)
!
GR2(1)=R(1)*PUXX+R(2)*PUYX+R(3)*PUZX
GR2(2)=R(1)*PUXY+R(2)*PUYY+R(3)*PUZY
GR2(3)=R(1)*PUXZ+R(2)*PUYZ+R(3)*PUZZ
!
!COMPUTE THIRD GRAD TERM, GR3(3)
!
GR3(1)=(R(1)*PUXX+R(2)*PUXY+R(3)*PUXZ)*(-1.0)
GR3(2)=(R(1)*PUYX+R(2)*PUYY+R(3)*PUYZ)*(-1.0)
GR3(3)=(R(1)*PUZX+R(2)*PUZY+R(3)*PUZZ)*(-1.0)
!
!COMPUTE U
!
DO 10 I=1,3
    U(I)=Co*(CR(I)+GR1(I)+GR2(I)+GR3(I))
10 CONTINUE
!
IF(N.EQ.2)THEN
    WRITE(7,11)CR
    WRITE(7,11)GR1

```

```
        WRITE(7,11)GR2
        WRITE(7,11)GR3
        WRITE(7,11)U
        END IF
    IF(N.EQ.3)THEN
        WRITE(6,11)CR
        WRITE(6,11)GR1
        WRITE(6,11)GR2
        WRITE(6,11)GR3
        WRITE(6,11)U
        WRITE(6,12)
        END IF
    IF(N.EQ.4)THEN
        WRITE(5,11)CR
        WRITE(5,11)GR1
        WRITE(5,11)GR2
        WRITE(5,11)GR3
        WRITE(5,11)U
        END IF
    !
11  FORMAT(3E14.7)
12  FORMAT("")
    CLOSE(7)
    CLOSE(6)
    CLOSE(5)
    RETURN
END
```



```

r1s = x1*x1+y1*y1+z1*z1
if (r1s .lt. beadsizesq) then
  goto 1000
endif

if (r1s .gt. sphere1/4.) then
  goto 1000
endif

1010 call random
x2 = scale1*(rand - 0.5)*0.5
call random
y2 = scale1*(rand - 0.5)*0.5
call random
z2 = scale1*(rand - 0.5)*0.5
c   x2=x2-x1
c   y2=y2-y1
c   z2=z2-z1

r12s = (x2)*(x2)+(y2)*(y2)+(z2)*(z2)

if (r12s .lt. beadsizesq) then
  goto 1010
endif

if(r12s.gt.sphere1/4.) then
  goto 1010
endif

x3=-x2
y3=-y2
z3=-z2

x4 = -x1
y4 = -y1
z4 = -z1

C Compute distances.  Impose bead size approach limit.

r4s = x4*x4+y4*y4+z4*z4

r1m = sqrt(r1s)

```

```

r2m = sqrt(x2*x2+y2*y2+z2*z2)
r3m = r2m
r4m = r1m
r1r2 = ((x1*x2)+(y1*y2)+(z1*z2))/r1m/r2m
R1R3 = ((x1*x3)+(y1*y3)+(z1*z3))/r1m/r3m
R2R3 = ((x2*x3)+(y2*y3)+(z2*z3))/r2m/r3m
R1R4 = ((x1*x4)+(y1*y4)+(z1*z4))/r1m/r4m
R2R4 = ((x2*x4)+(y2*y4)+(z2*z4))/r2m/r4m
R3R4 = ((x3*x4)+(y3*y4)+(z3*z4))/r3m/r4m
!
!COMPUTE A
!
COA=27.0/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)
T1A=(1.0-3.0*R1R2*R1R2)*(1.0-3.0*R2R3*R2R3)
T2A=6.0*R1R2*R1R3*R2R3
T3A=6.0*R1R2*R1R2*R2R3*R2R3
Aang = T1A-T2A+T3A
A = CoA * Aang

!FINALLY, MAYBE I HAVE AN EXPRSSION FOR U4
!
COU4=(-3.0)/4.0/(R4M*R4M)
COTXU4=(-1.0)*81.0/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)
POL13=R1R3-R1R2*R2R3
POL14=R1R4-R1R2*R2R4
POL24=R2R4-R2R3*R3R4
POL24A=R2R4-5.0*R2R3*R3R4
T2U4=T1A*R3R4*R3R4
T3U4=2.0*R1R2*POL13*POL24A*R3R4
T4U4=2.0*(1.0-3.0*R1R2*R1R2)*R2R3*POL24*R3R4
T5U4=2.0*R1R2*R2R3*POL14*R3R4
TXang=T2U4+T3U4+T4U4+T5U4
value1 = R1R4*(COU4*CoA*Aang+CoU4*COTXU4*TXang)
nintegratecount1 = nintegratecount1 + 1
total1 = total1 + value1

```

```

C fill second box
C Return here if positions are not good.
1001 continue
call random
x1 = scale2*(rand - 0.5)*0.5
call random
y1 = scale2*(rand - 0.5)*0.5

```

```

call random
z1 = scale2*(rand - 0.5)*0.5

r1s = x1*x1+y1*y1+z1*z1

if (r1s .lt. beadsizesq) then
  goto 1001
endif

if (r1s .gt. sphere2/4.) then
  goto 1001
endif

1020 call random
x2 = scale2*(rand - 0.5)*0.5
call random
y2 = scale2*(rand - 0.5)*0.5
call random
z2 = scale2*(rand - 0.5)*0.5

c    x2=x2-x1
c    y2=y2-y1
c    z2=z2-z1

r12s = (x2)*(x2)+(y2)*(y2)+(z2)*(z2)

if (r12s .lt. beadsizesq) then
  goto 1020
endif

if (r12s.gt.sphere2/4.) then
  goto 1020
endif

x3=-x2
y3=-y2
z3=-z2
x4 = -x1
y4 = -y1
z4 = -z1

```

C Compute distances. Impose bead size approach limit.

```

r1m = sqrt(r1s)
r2m = sqrt(x2*x2+y2*y2+z2*z2)
r3m = r2m
r4m = r1m
r1r2 = ((x1*x2)+(y1*y2)+(z1*z2))/r1m/r2m
R1R3 = ((x1*x3)+(y1*y3)+(z1*z3))/r1m/r3m
R2R3 = ((x2*x3)+(y2*y3)+(z2*z3))/r2m/r3m
R1R4 = ((x1*x4)+(y1*y4)+(z1*z4))/r1m/r4m
R2R4 = ((x2*x4)+(y2*y4)+(z2*z4))/r2m/r4m
R3R4 = ((x3*x4)+(y3*y4)+(z3*z4))/r3m/r4m
!
!COMPUTE A
!
COA=27.0/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)
T1A=(1.0-3.0*R1R2*R1R2)*(1.0-3.0*R2R3*R2R3)
T2A=6.0*R1R2*R1R3*R2R3
T3A=6.0*R1R2*R1R2*R2R3*R2R3
Aang = T1A-T2A+T3A
A = CoA * Aang

!FINALLY, MAYBE I HAVE AN EXPRSSION FOR U4
!
COU4=(-3.0)/4.0/(R4M*R4M)
COTXU4=(-1.0)*81.0/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)
POL13=R1R3-R1R2*R2R3
POL14=R1R4-R1R2*R2R4
POL24=R2R4-R2R3*R3R4
POL24A=R2R4-5.0*R2R3*R3R4
T2U4=T1A*R3R4*R3R4
T3U4=2.0*R1R2*POL13*POL24A*R3R4
T4U4=2.0*(1.0-3.0*R1R2*R1R2)*R2R3*POL24*R3R4
T5U4=2.0*R1R2*R2R3*POL14*R3R4
TXang=T2U4+T3U4+T4U4+T5U4
value2 = R1R4*(COU4*CoA*Aang+CoU4*COTXU4*TXang)

```

```

C Test if both particles within inner sphere
  iflag1 = 1
  if (r1s .gt. sphere1/4..and. r12s.le.sphere1/4.) then
    iflag1 = 0
    nintcount2a = nintcount2a + 1
    total2a = total2a + value2

```

```

endif
if (r12s .gt. sphere1/4..and.r1s.le.sphere1/4.) then
    iflag1 = 0
    nintcount2b = nintcount2b + 1
    total2b = total2b + value2
endif
if (r1s.gt.sphere1/4..and.r12s.gt.sphere1/4.) then
    iflag = 0
    nintcount2c = nintcount2c + 1
    total2c = total2c + value2
endif
if (r1s.le.sphere1/4..and.r12s.le.sphere1/4.) then
    nintegratecount2a=nintegratecount2a + 1
    total2d = total2d+value2
endif

```

C fill third box

C Return here if positions are not good.

```

1002 continue
    call random
    x1 = scale3*(rand - 0.5)*0.5
    call random
    y1 = scale3*(rand - 0.5)*0.5
    call random
    z1 = scale3*(rand - 0.5)*0.5

    r1s = x1*x1+y1*y1+z1*z1

    if (r1s .lt. beadsizesq) then
        goto 1002
    endif

    if (r1s .gt. sphere3/4.) then
        goto 1002
    endif

1030 call random
    x2 = scale3*(rand - 0.5)*0.5
    call random
    y2 = scale3*(rand - 0.5)*0.5
    call random

```

```

z2 = scale3*(rand - 0.5)*0.5

c      x2=x2-x1
c      y2=y2-y1
c      z2=z2-z1

r12s = (x2)*(x2)+(y2)*(y2)+(z2)*(z2)

if (r12s .lt. beadsizesq) then
  goto 1030
endif

if (r12s.gt.sphere3/4.) then
  goto 1030
endif

x3=-x2
y3=-y2
z3=-z2
x4 = -x1
y4 = -y1
z4 = -z1

C Compute distances.  Impose bead size approach limit.

r1m = sqrt(r1s)
r2m = sqrt(x2*x2+y2*y2+z2*z2)
r3m = r2m
r4m = r1m
r1r2 = ((x1*x2)+(y1*y2)+(z1*z2))/r1m/r2m
R1R3 = ((x1*x3)+(y1*y3)+(z1*z3))/r1m/r3m
R2R3 = ((x2*x3)+(y2*y3)+(z2*z3))/r2m/r3m
R1R4 = ((x1*x4)+(y1*y4)+(z1*z4))/r1m/r4m
R2R4 = ((x2*x4)+(y2*y4)+(z2*z4))/r2m/r4m
R3R4 = ((x3*x4)+(y3*y4)+(z3*z4))/r3m/r4m
!
!COMPUTE A
!
COA=27.0/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)
T1A=(1.0-3.0*R1R2*R1R2)*(1.0-3.0*R2R3*R2R3)
T2A=6.0*R1R2*R1R3*R2R3
T3A=6.0*R1R2*R1R2*R2R3*R2R3

```

```

Aang = T1A-T2A+T3A
A = CoA * Aang

!FINALLY, MAYBE I HAVE AN EXPRSSION FOR U4
!
COU4=(-3.0)/4.0/(R4M*R4M)
COTXU4=(-1.0)*81.0/64.0/(R1M*R1M)/(R2M*R2M*R2M)/(R3M*R3M*R3M)
POL13=R1R3-R1R2*R2R3
POL14=R1R4-R1R2*R2R4
POL24=R2R4-R2R3*R3R4
POL24A=R2R4-5.0*R2R3*R3R4
T2U4=T1A*R3R4*R3R4
T3U4=2.0*R1R2*POL13*POL24A*R3R4
T4U4=2.0*(1.0-3.0*R1R2*R1R2)*R2R3*POL24*R3R4
T5U4=2.0*R1R2*R2R3*POL14*R3R4
TXang=T2U4+T3U4+T4U4+T5U4
value3 = R1R4*(COU4*CoA*Aang+CoU4*COTXU4*TXang)

```

```

C Test if both particles within inner sphere
iflag1 = 1
if (r1s .gt. sphere2/4..and.r12s.le.sphere2/4.) then
    iflag1 = 0
    nintcount3a = nintcount3a + 1
    total3a = total3a + value3
endif
if (r12s .gt. sphere2/4..and.r1s.le.sphere2/4.) then
    iflag1 = 0
    nintcount3b = nintcount3b + 1
    total3b = total3b + value3
endif
if (r1s .gt. sphere2/4..and.r12s.gt.sphere2/4.) then
    iflag1 = 0
    nintcount3c =nintcount3c + 1
    total3c = total3c + value3
endif
if (r1s.le.sphere2/4..and.r12s.le.sphere2/4.) then
    nintegratecount3a=nintegratecount3a + 1
    total3d = total3d+value3
endif

```

```

2000 continue
return

```



```

vol3 = scale3*scale3*scale3*3.14159/48.0

if (nintegratecount3a .gt. 0) then
  total3d = total3d/nintegratecount3a*(vol2-vol0)**2
endif

if (nintcount3a.gt.0) then
  total3a = total3a/nintcount3a
  testtotal3a = total3a
  total3a = total3a * (vol2-vol0) * (vol3-vol2)
endif

if (nintcount3b.gt.0) then
  total3b = total3b/nintcount3b
  testtotal3b = total3b
  total3b = total3b * (vol2-vol0) * (vol3-vol2)
endif

if (nintcount3c.gt.0) then
  total3c = total3c/nintcount3c
  testtotal3c = total3c
  total3c = total3c * ((vol3-vol2)**2)
endif

P1 = (vol1-vol0)/(vol2-vol0)
Psh = (vol2-vol1)/(vol2-vol0)
P2 = (vol2-vol0)/(vol3-vol0)
P3sh = (vol3-vol2)/(vol3-vol0)

total2 = (P1*Psh*(total2a+total2b)+Psh*Psh*total2c)/
Q (2.*P1*Psh+Psh*Psh)

total3 = (P2*P3sh*(total3a+total3b)+P3sh*P3sh*total3c)/
Q (2.*P2*P3sh+P3sh*P3sh)

total = total1+total2a+total2b+total2c+
Q total3a+total3b+total3c

analy1 =- 16.*3.141592654**2*((1./beadsizes-4./scale1)*
Q 1./beadsizes**3-(4./scale1)**3)/3.)*(6./5.)*(81./128.)

analy2 =-(((1./beadsizes-4./scale1)*((4./scale1)**3-(4./scale2)

```

```

Q **3)/3.+(4./scale1-4./scale2)*(1/beadsizes**3-(4./scale1)
Q **3)/3)*P1*Psh+(4./scale1-4./scale2)*((4./scale1)**3
Q -(4./scale2)**3)/3.*Psh*Psh)
Q *16*3.141592654**2/(Psh*Psh+2*P1*Psh)*(6./5.)*(81./128.)

analy2a =- 16.*3.141592654**2*
Q (1./beadsizes**3-(4./scale1)**3)*
Q (4./scale1-4./scale2)/3.*(6./5.)*(81./128.)

analy2b = -16.*3.141592654**2*
Q (1./beadsizes-4./scale1)*
Q ((4./scale1)**3-(4./scale2)**3)/3.*(6./5.)*(81./128.)

analy2c = -16.*3.141592654**2*
Q (4./scale1-4./scale2)*
Q ((4./scale1)**3-(4./scale2)**3)/3.*(6./5.)*(81./128.)

analy3 =-(((1/beadsizes-4./scale2)*((4./scale2)**3-(4./scale3)
Q **3)/3.+(4./scale2-4./scale3)*(1/beadsizes**3-(4./scale2)
Q **3)/3.)*P2*P3sh+(4./scale2-4./scale3)*((4./scale2)**3
Q -(4./scale3)**3)/3.*P3sh*P3sh)
Q *16*3.141592654**2/(P3sh*P3sh+2.*P2*P3sh)*(6./5.)*(81./128.)

analytic = -(1./beadsizes)*(1/(beadsizes**3))*6./5.*81./128.*
Q 16.*3.141592654**2/3.

analy3a = -16.*3.141592654**2*
Q (1./beadsizes**3-(4./scale2)**3)*
Q (4./scale2-4./scale3)/3.*(6./5.)*(81./128.)

analy3b = -16.*3.141592654**2*
Q (1./beadsizes-4./scale2)*
Q ((4./scale2)**3-(4./scale3)**3)/3.*(6./5.)*(81./128.)

analy3c = -16.*3.141592654**2*
Q (4./scale2-4./scale3)*
Q ((4./scale2)**3-(4./scale3)**3)/3.*(6./5.)*(81./128.)

analytot = analy1+analy2a+analy2b+analy2c+analy3a+
Q analy3b+analy3c

analytwo = analy2a+analy2b+analy2c

```

```

analythree = analy3a+analy3b+analy3c

nintegratecount2 = nintcount2a+nintcount2b+nintcount2c
nintegratecount3 = nintcount3a+nintcount3b+nintcount3c

open(63, file = 'flyI.dat')

write(63,*) runname$
write(63,*) 'scale1,scale2,scale3,beadsizes,nintegrate'
write(63,*) scale1,scale2,scale3,beadsizes,nintegrate

write(63,*) 'total1,nintegrate,nintegratecount1'
write(63,*) total1,nintegrate,nintegratecount1
write(63,*) 'analytic 1',analy1
write(63,*) 'total', total
write(63,*)
write(63,*) 'total2,nintegratecount2'
write(63,*) total2,nintegratecount2
write(63,*) 'total2a,b,c'
write(63,*) total2a,total2b,total2c
write(63,*) 'analy2a,analy2b,analy2c',analy2a,analy2b,analy2c
write(63,*) 'analytic 2',analy2
write(63,*)
write(63,*) 'total2d,nintegratecount2a'
write(63,*) total2d,nintegratecount2a
write(63,*) 'total3,nintegratecount3'
write(63,*) total3,nintegratecount3
write(63,*) 'total3a,b,c'
write(63,*) total3a,total3b,total3c
write(63,*) 'analy3a,analy3b,analy3c',analy3a,analy3b,analy3c
write(63,*) 'analytic 3',analy3
write(63,*)
write(63,*) 'total3d,nintegratecount3a'
write(63,*) total3d,nintegratecount3a

close (63, status = 'KEEP')

open (63, file = 'comresults.dat')
write(63,*) 'beadsizes,beadsizes,nintegrate',nintegrate
write(63,*)
write(63,*) 'scale, analytic, numeric, counts, error'

```



```

read(63,*) a$
read(63,*) scale3

read(63,*) a$
read(63,*) beadsize

read(63,*) a$
read(63,*) nintegrate

read(63,*) a$
read(63,*) idum

close (63, status = 'KEEP')
beadsizesq = beadsize*beadsize
sphere1= scale1*scale1 / 4.
sphere2= scale2*scale2 / 4.
sphere3= scale3*scale3 / 4.

seed = idum

```

C finished reading control file; close the file.
 close (63, status = 'KEEP')

C Initialize variables requiring initialization.

```

value1 = 0.00000000000000000000
value2 = 0.00000000000000000000
value3 = 0.00000000000000000000
total1 = 0.00000000000000000000
total2 = 0.00000000000000000000
total3 = 0.00000000000000000000
total2a = 0.00000000000000000000
total2b = 0.00000000000000000000
total2c = 0.00000000000000000000
total2d = 0.00000000000000000000
total3a = 0.00000000000000000000
total3b = 0.00000000000000000000
total3c = 0.00000000000000000000
total3d = 0.00000000000000000000
nintegratecount1= 0

nintcount2a = 0

```



```
nintcount2b = 0  
nintcount2c = 0  
nintegratecount2a= 0
```

```
nintcount3a = 0  
nintcount3b = 0  
nintcount3c = 0  
nintegratecount3a= 0
```

```
return  
end
```

Appendix D

flyintegrat.com

```
C *****
C * COMMON BLOCKS flyintegrat.com *
C * common blocks and variables for integrate *
C * GDJP June 1997 *
C *****

C      implicit undefined (a-z)
C An error trap

C common block for integer variables
      common/integervars/ iterate,iintegrate,nintegrate,iflag1,
Q      nintegratecount1,nintegratecount2,nintegratecount3,
Q      nintegratecount1a,nintegratecount2a,nintegratecount3a,
Q      nintcount2a,nintcount2b,nintcount2c,
Q      nintcount3a,nintcount3b,nintcount3c

      integer*4      iterate,iintegrate,nintegrate,iflag1,
Q      nintegratecount1,nintegratecount2,nintegratecount3,
Q      nintegratecount1a,nintegratecount2a,nintegratecount3a,
Q      nintcount2a,nintcount2b,nintcount2c,
Q      nintcount3a,nintcount3b,nintcount3c

C*****
C common block for random number generator
      common/randomvars/facg,gasdevg,gsetg,inull,iset3g,rand,temp,
Q      temprand,v1g,v2g,xrandom,yrandom,zrandom,idum,seed

      integer*4 inull,iset3g, xrandom, yrandom, zrandom,idum,seed
```

```

real*4 facg,gasdevg,gsetg,rand,rg,temprand,temp,v1g,v2g

C common block for real dynamic variables
  common/realvars/
Q  x1,y1,z1,r1,x2,y2,z2,r2,r12,scale,beadsizesq,
Q  r1s,r2s,r12s,
Q  scale1,scale2,scale3,sphere1,sphere2,sphere3,
Q  ave,vol0,vol1,vol2,vol3,
Q  value1,value2,value3,
Q  total1,total2,total2a,total3,total3a,
Q  total2b,total3c,
Q  total2c,total2d,total3b,total3d,
Q  analy2,analy2a,analy2b,analy3,analy3a,analy3b

real*8
Q  x1,y1,z1,r1,x2,y2,z2,r2,r12,scale,beadsizesq,
Q  r1s,r2s,r12s,
Q  scale1,scale2,scale3,sphere1,sphere2,sphere3,
Q  ave,vol0,vol1,vol2,vol3,
Q  value1,value2,value3,
Q  total1,total2,total2a,total3,total3a,
Q  total2b,total3c,
Q  total2c,total2d,total3b,total3d,
Q  analy2,analy2a,analy2b,analy3,analy3a,analy3b

C*****
C common block for timestamp variables
  common/timestamp/hrs,hundrs,mins,secs,time1,time2

integer*4 hrs,hundrs,mins,secs

real*4 time1,time2

C*****
  common/loosechars/a$,a1$,a2$,name$,runname$
character          a$,a1$,a2$,name$,runname$

c      dimension drij(3),force(100,3),
c      Q    rsphere(100,3),rij(3)

```

Bibliography

- [1] G.D.J.Phillies. *Macromolecules* **31**(98)2317-2327.
- [2] J.G.Kirkwood and Riseman, J. *J.Chem.Physics.* **16**(48)565-573.
- [3] H.Staudinger. *Die Hochmolecularen Organischen Verbindungen*(Verlag Julius Springer, Berlin, 1932).
- [4] W.Kuhn. *Zeits. f. physik Chemie* **A161**(32)1.
- [5] M.L.Huggins. *J.Phys.Chem.* **43**(39)439.
- [6] H.A.Kramers. *J.Chem.Phys.* **14**(46)415.
- [7] P.Debye. Reports to the Office of Rubber Reserve, Reconstruction Finance Corp, October 10, 1946.
- [8] P.Debye. *Phys.Rev.* **71**(47)486.
- [9] H.C.Brinkman. *Physica* **13**(47)447.
- [10] M.Adam and M.Delsanti. *J.Colloid Interface Sci.* **10**(77)1229.
- [11] G.D.J.Phillies. In press.