Mathematical Modelling of Hybrid Photonic Structures for Holographic Sensors

Jack Lyons

Supervisor: Dr. Dana Mackey

Advisory Supervisor: Prof. Izabela Naydenova

PhD Thesis submitted to the School of Mathematics & Statistics, Technological University Dublin

2024

Declaration

I hereby declare that this thesis is entirely my own original work. All other sources of information presented which are not part of my work have been properly cited. This thesis was prepared according to the regulations for postgraduate studies by research of Technological University Dublin and has not been submitted for any other degree at any other Institute or University. This work reported on this thesis conforms to the principle and requirements of Technological University Dublin's guidelines for ethics in research. The University has permission to keep, or lend or to copy this thesis in whole or in part, on condition that use the material or thesis be duly acknowledged.

Jack Lyons

2024

Acknowledgements

Abstract

This thesis outlines a mathematical framework for modelling the formation of holographic gratings in hybrid photopolymer based nanocomposites with the aim of optimising their holographic recording properties for optical sensing applications. Thus, the second aim of the work is to model the change in optical properties of the grating in response to exposure to a target analyte. This work has been a collaborative research project between the School of Mathematics & Statistics at Technological University Dublin and the Centre for Industrial and Engineering Optics that have done extensive experimental work with holographic gratings recorded in photopolymer materials.

In recent years, there has been increased interest towards hybrid photopolymers, experimental researchers at TU Dublin have been particularly interested in acrylamide-based photopolymers doped with zeolite nanoparticles. The reasons for this interest are threefold: first, nanocomposites allow for greater flexibility in achieving the desired material properties since one can tune the properties of the different components independently; second, it has been demonstrated in the literature that the incorporation of nanoparticles in photopolymers reduces the amount of shrinkage of which they normally suffer; third, by adding nanoparticles it is possible to functionalise the materials and the structures recorded in them and thus use them in, for example, optical sensing. Previously reported results from experiments with the new hybrid holographic materials doped with nanozeolites have revealed some positive features: reduced photopolymerization-induced material shrinkage and a non-linear increase in refractive index modulation in response to increased concentration of inorganic nanoparticles which existing mathematical models are not capable of predicting.

This thesis reviews the existing mathematical models and builds on them by incorporating: new equations to describe the spatial redistribution of inorganic nanoparticles during holographic recording; cross-diffusion in the mass transport of monomer and short polymer chains; scattering of light from inert nanoparticles; Bragg angle detuning due to shrinkage during holographic recording of slanted gratings; and absorption of light within the finite depth of the nanocomposite. Where earlier models restricted analysis to the distance across the grating period, the last two aforementioned improvements require analysis to be extended to the depth within the nanocomposite. The model is validated via a comparison of numerical simulations of the model equations with existing experimental data. The model has successfully

predicted the change in refractive index modulation in response to increased doping of inorganic nanoparticles and the increased shrinkage at high spatial frequencies. The results predicting the effect of the nanoparticle refractive index on the final refractive index modulation of the grating are very useful for guiding the design of holographic recording materials. This is particularly important where high diffraction efficiency optical structures are needed in thinner polymeric layers. In the context of environmental sensors, this would ensure a faster response time and larger operational angles.

Furthermore, the final condition of this model is used as the initial condition for a second mathematical framework modelling the diffusion of target analytes in a holographic grating and the resulting change in optical properties. The end product of this research is a mathematical model which can offer some insight into how properties of the photopolymer, the recording conditions and nanoparticle properties can be used to optimize holographic gratings recorded in hybrid photopolymer materials for the purpose of holographic sensor functionality by quantifying the refractive index modulation sensitivity to nanoparticle concentration and minimizing both distortion and shrinkage. A systematic approach to the fabrication of novel hybrid materials based on mathematical and computational modelling will significantly shorten the design phase. The analytic characterization of such materials can be translated into practical recommendations, leading to direct and immediate impact on the applied research in holographic sensors.

Table of contents

List of figures

List of tables

Nomenclature

Roman Symbols

$\Delta\phi_r$ — Rotation of the slant angle

Δn — Rrefractive index modulation

δ — Distortion

ϵ_{ij} — Cross-diffusion constant

η — Diffraction efficiency

Γ — Immobilization constant

γ_s — Capture coefficient for analyte

Λ — Grating period

λ_r — Wavelength of recording wavefront

ω_s — Escape coefficient for analyte

$\overline{n}$ — Average refractive index

ϕ — Unrefracted slant angle

ϕ_r — Refracted slant angle

ρ' — Moharam-Young Criteria

ρ_i — Density of component i

τ_c — Characteristic time of analyte capture

τ_e — Characteristic time of analyte escape

τ_x — Characteristic time of analyte diffusion (in x-direction)

τ_y — Characteristic time of analyte diffusion (in y-direction)

θ_B — Bragg angle

ξ — Scattering coefficient

ζ Absorption Coefficient

b_0 Initial mass concentration of binder

D_s Diffusion coefficient of analyte

F_0 Polymerization constant

I_0 Intensity of recording beam

J_m Bessel function of order m

J_m Diffusion flux of monomer

J_p Diffusion flux of mobile short polymers

J_z Diffusion flux of inorganic nanoparticles

m Mass concentration of monomer

m_0 Initial mass concentration of monomer

$n_z^{(e)}$ Refractive index of porous nanoparticle filled with partial vacuum

$n_z^{(s)}$ Refractive index of porous nanoparticle filled with analyte molecules

n_i Refractive index of component i

p Mass concentration of short polymer

q Mass concentration of immobilized polymer

Q' Klein-Cook Criteria

s Mass concentration of analyte

T Thickness of holographic material

V_0 Visibility of illumination pattern

z Mass concentration of inorganic nanoparticles

$z^{(e)}$ Vacant nanozeolites

$z^{(s)}$ Occupied nanozeolites

z_0 Initial mass concentration of inorganic nanoparticles

AA   Acrylamide

BA   NN'methylenebisacrylamide

IEO   Centre for Industrial and Engineering Optics

NPDD Non-local polymerization driven diffusion

PDE   Partial Differential Equation

PVA   Polyvinyl alcohol

RI   Refractive index

SRG   Surface Relief Grating

Chapter 1

Introduction

1.1 Background

Reflection, refraction and diffraction are the three mechanisms by which the path of a beam of light can be changed. Most people will be more familiar with the first two phenomena which, for centuries, have formed the basis for the design of optical instruments that have facilitated the study of stars beyond our own galaxy and of microorganisms smaller than a grain of salt. It was Grimaldi who coined the word 'diffraction' from the Latin *to break into pieces* in the 17th century to describe the patterns of alternating bright and dark bands he observed. At the time, the classical (or corpuscular) theory viewed light as a stream of particles emitted from a source, this was adequate to describe reflection and refraction but could never provide a satisfactory explanation of Grimaldi's observations. In 1803 Young's double-slit experiment demonstrated the diffraction of light and how the behaviour can be predicted when light is treated as a wave. The classical theory gave way to the wave propagation theory of light. Huygens and Fresnel provided a more complete understanding of diffraction with their principle which stated that every point in a wavefront is a source of secondary wavelets, the superposition of wavelets originating from different points forms a new wavefront. Today the insights of Planck [9] and Einstein [10] have reconciled the classical theory and the wave propagation theory into the quantum theory of light but the wave description of light allowed physicists to better understand a further phenomenon: the interference of light waves. The principle of superposition of waves states that when two or more waves are incident at the same point in space, the amplitude of the resultant wave is equal in magnitude to the amplitude of the vector sum of the individual waves. For example consider two optical waves with electric field magnitude E_1 and E_2, and phase $\vec{k}_1 \cdot \vec{r} - \omega_1 t$ and $\vec{k}_2 \cdot \vec{r} - \omega_2 t$, the intensity of each wave can be expressed as a function of its position $\vec{r}$ and time t

$$I_1(\vec{r}, t) = \left\langle \left[E_1 \cos(\vec{k}_1 \cdot \vec{r} - \omega_1 t) \right]^2 \right\rangle = \frac{1}{2} E_1^2,$$

$$I_2(\vec{r}, t) = \left\langle \left[E_2 \cos(\vec{k}_2 \cdot \vec{r} - \omega_2 t) \right]^2 \right\rangle = \frac{1}{2} E_2^2,$$

where $\vec{k}_1$ and $\vec{k}_2$ are the respective wave vectors, ω_1 and ω_2 are the frequencies and the intensity of the resultant wave from their interference can be expressed as

$$
\begin{aligned}
I(\vec{r}, t) &= \left\langle \left[E_1 \cos(\vec{k}_1 \cdot \vec{r} - \omega_1 t) + E_2 \cos(\vec{k}_2 \cdot \vec{r} - \omega_2 t) \right]^2 \right\rangle \\
&= \frac{1}{2} E_1^2 + \frac{1}{2} E_2^2 + \langle 2 E_1 E_2 \cos(\vec{k}_1 \cdot \vec{r} - \omega_1 t) \cos(\vec{k}_2 \cdot \vec{r} - \omega_2 t) \rangle \\
&= I_1 + I_2 + 2\sqrt{I_1 I_2} \left\{ \langle \cos \left[(\vec{k}_1 + \vec{k}_2) \cdot \vec{r} - (\omega_1 + \omega_2) t \right] \rangle \right. \\
&\left. + \langle \cos \left[(\vec{k}_1 - \vec{k}_2) \cdot \vec{r} - (\omega_1 - \omega_2) t \right] \rangle \right\}
\end{aligned}
$$

The expression above shows that if two interfering waves are oscillating at the same frequency (coherent waves) then the equation for the intensity of the resultant wave simplifies significantly; if the total intensity of the two optical waves is, $I_1 + I_2 = I_0$, V_0 is the visibility and $\vec{K}_g$ the grating vector then the resultant wave intensity is described as

$$\frac{I(\vec{r})}{I_0} = 1 + \frac{2\sqrt{I_1 I_2}}{I_1 + I_2} \cos \left[(\vec{k}_1 - \vec{k}_2) \cdot \vec{r} \right] = 1 + V_0 \cos \left(\vec{K}_g \cdot \vec{r} \right) \tag{1.1}$$

The visibility (or contrast) of the interference pattern takes a maximum value of 1 when the intensities of the two optical waves are equal; the grating vector is determined by the angle between the two waves, its magnitude is related to the distance between two adjacent bright fringes Λ, better known as the grating period.

$$\left| \vec{K}_g \right| = \left| \vec{k}_1 - \vec{k}_2 \right| = \frac{2\pi}{\Lambda}.$$

The grating vector direction will be normal to the parallel planes of intensity maxima, also known as the Bragg planes. The direction of diffraction maxima is defined by Bragg's law which tells us that constructive interference from two interfering plane waves occurs when the path difference between wavefronts is equal to an integer number of wavelengths. If λ_r is the wavelength of the source, n the refractive index (RI) of the material and ϕ is the angle between the grating vector and the surface normal vector (the slant angle) then the Bragg angle θ_B is defined by [3]

$$\sin(\phi + \theta_B) = m \frac{\lambda_r}{2n\Lambda}. \tag{1.2}$$

Both angles and their relationship to the vectors are defined in Fig. 1.1.
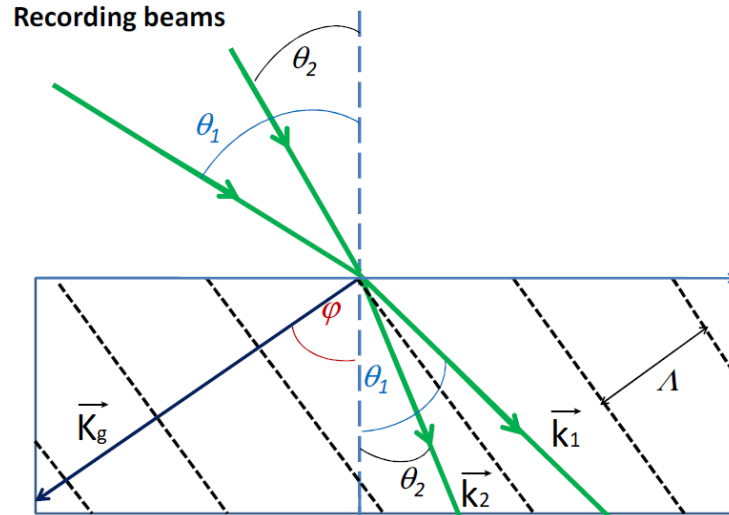
Fig. 1.1 Geometry of two interfering coherent plane waves, [1].

It's possible to create a permanent record of this interference pattern in the form of a spatial modulation of optical properties such as RI or absorption coefficient, the record is called a holographic grating. Holography is distinguished from photography in that the latter records only the amplitude of the optical wave whereas the former records both the amplitude and phase. A holographic grating is characterized by the RI modulation Δn defined as the difference between the maximum and minimum values of RI. There are two classifications into which holographic gratings may be assigned: thick and thin. The names are not entirely appropriate as the classification is based not on the physical thickness of the material but rather on the energy distribution around the first diffraction order. Thick and thin regimes display distinctly different behaviour and require different theories to describe the distribution of energy in diffracted waves. The former is described by Kogelnik's coupled wave theory [11], while the latter is described by Raman-Nath theory [12, 13, 14]. Thick holographic gratings are sometimes called volume regime gratings because the Bragg planes must extend inside the volume of the recording media. In recording media with greater depth the incident beam interacts with more planes of periodically varying RI and thus more energy is transferred to the diffracted beam. By contrast, in a recording media with minimal depth there is less opportunity for light-matter interaction and more energy goes to the higher diffraction orders. Thin holographic gratings are better known as plane phase holographic gratings and are characterised by multiple visible diffraction orders; volume phase gratings have only single diffraction orders (Fig. 1.2).

(a) Plane   (b) Volume
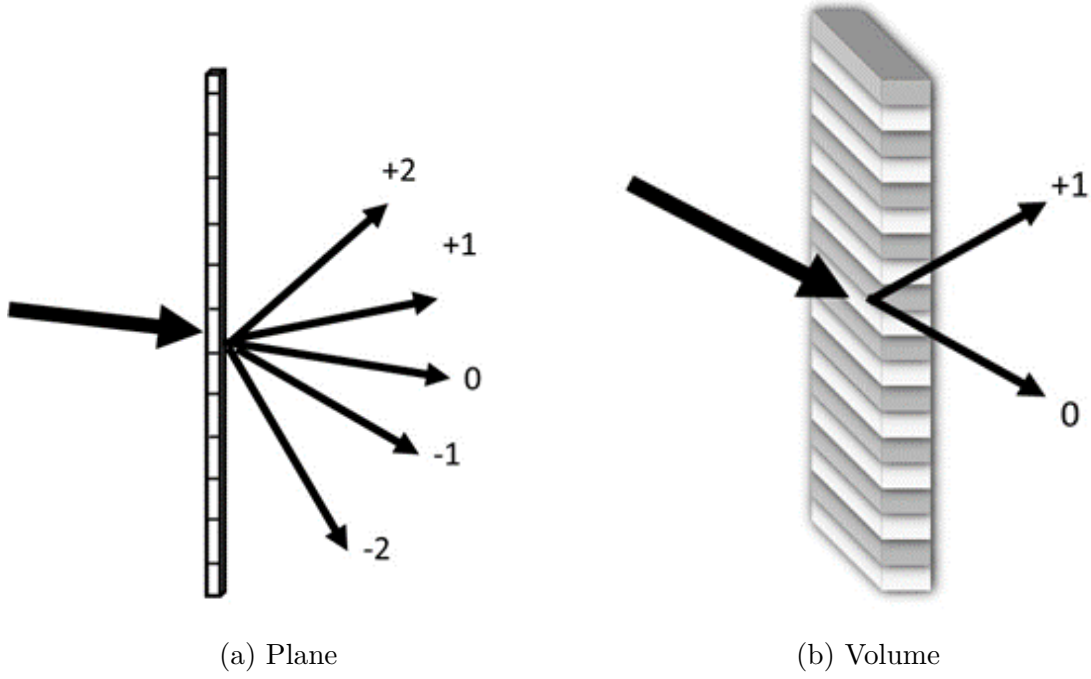
Fig. 1.2 Plane and volume holographic gratings.

Volume gratings are advantageous for some holographic applications such as imaging and sensors because most of the diffracted energy is in the first order. In terms of holographic applications, transmitted energy in higher diffraction orders is noise in the received signal and reduces the efficiency of the first order. The commercial interest in applications of volume holographic gratings creates a need for a well defined boundary that divides the volume and planar regimes, through a combination of experimental observation and numerical approximations of the coupled-wave theory various criteria have emerged, the first reliable being the Klein-Cook criteria [15] which states that for a diffraction grating recorded in a sample with thickness T,

$$Q' = \frac{2\pi\lambda_r T}{n\Lambda^2}.$$

$Q' < 10$ for plane phase gratings and $Q' > 10$ for volume phase gratings. In materials with very large RI modulation the Klein-Cook criteria deteriorates in its ability to accurately predict the behaviour of holographic gratings, in such cases a more reliable condition is the Moharam-Young criteria [16],

$$\rho' = \frac{\lambda_r^2}{n\Delta n\Lambda^2},$$

$\rho' < 1$ for plane phase gratings and $\rho' > 1$ for volume phase gratings; note that the physical thickness of the sample does not factor in the criteria.

There is a range of different permanent holographic recording media (materials in which the recorded diffraction pattern cannot be erased or altered) available. For

example, colloidal suspension of crystals such as AgBr or AgCl in gelatin can be induced by an illumination pattern, postexposure processing of the silver halide emulsion can reconstruct the pattern [3]; this process is similar to how silver halides are used in photography, the difference is that the crystals need to be significantly smaller (in the order of nm as opposed to μm for photography), the bleaching process is also different for the holographic material so as to transform amplitude modulation to phase modulation [17, 3]. Another gelatin based holographic material is dichromated gelatin, the use of ammonium dichromate rather than silver halides creates a holographic grating more limited in terms of sensitivity and spectral response though the materials are easier to procure and hence more cost effective to manufacture [3, 18]. The ability of photoresist materials to dissolve in solvent change when exposed to illumination make them a suitable media for surface relief gratings and are commonly used as a master for embossed hologram lithography [3, 19]. Inorganic glasses, e.g. Na_2O-ZnO-Al_2O_3-SiO_2 doped with small quantities of silver or cerium, are transparent in the visible spectrum but exposure to UV light followed by thermal precipitation of crystalline phase results in a drop in RI, such materials are called photo-thermo-refractive glasses are a suitable media for recording volume holographic gratings [3, 20]. This research will focus on the formation of holographic gratings fabricated using photopolymer materials.

1.2 Photopolymers

A photopolymer is made of organic molecules which can bind together to form solid layers upon exposure to visible light. The process by which the organic molecules (monomer) bind together is known as polymerization; short chains up to only a few dozen monomer in length are called oligomer, oligomer continue to grow to longer polymer chains which can exceed a few thousand monomer in length. Polymerization can be broken into three stages: initiation, propagation and termination; there is also cross-linking, a separate process. The basic formula for a photopolymer system is

1. Photoinitiating system: made up of two molecules, a sensitizing dye and an electron donor, e.g. Erytrosin B and Triethanolamine respectively. The dye absorbs a photon to promote it to a excited state and reacts with an electron donor to produce free radicals. Most free radicals will return to the dye but some will bind monomer together to form larger and less mobile oligomer, this is the initiation stage. Some early examples of photoinitiators developed in the 1960's operated in the UV spectrum, the development of dye-based photoinitiators allowed photopolymerization using visible light.

2. A monomer: mobile organic molecules, e.g. acrylamide (AA), react with free radicals binding together to form oligomer (short mobile polymers); the sustained growth of oligomer via the addition of additional radicalized monomer

is the propagation stage; when free-radicals interact with the ends of the chain preventing further propagation and chain growth, this is the termination stage.

3. A cross-linking monomer, e.g. NN'methylenebisacrylamide (BA): reacts with free radicals to bind oligomers together, amalgamating them to much larger, densely packed immobile polymer chains.

4. A binder matrix: Non-reactive liquid solvent, e.g. polyvinyl alcohol (PVA), makes up the bulk of the photosensitive layer's mass.

The rate of growth of polymer chains within some localized area of a sample is directly proportional to the number of photons falling in that area per unit time, i.e. the intensity. Because the intensity of the resultant wave formed by the interference of two optical waves is sinusoidal, the spatial profile of polymer chains will also be sinusoidal (Fig. 1.3). The concentration gradient that arises from exposing a photopolymer material to an illumination pattern results in a mass transport of system components and hence a change in the local density of monomer and polymer. Cross-linked polymer chains are immobile so the resulting physical structure will remain after the illumination is switched off, the resulting nanostructure is a holographic grating.
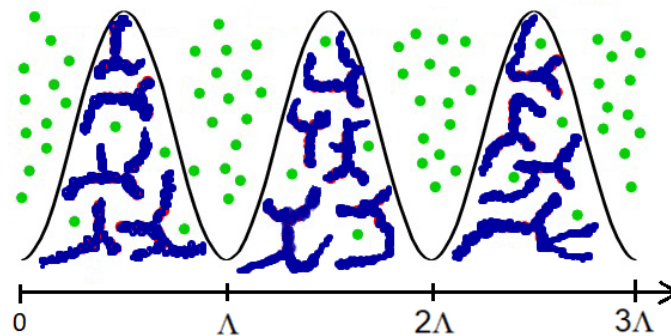
Fig. 1.3 Formation of a holographic grating on a photopolymer material exposed to sinusoidal illumination [2]; polymer chains (blue) form at the bright fringes, diffusion of monomer (green) allows continued polymerization.

The first hologram was produced in 1948 by Dennis Gabor, while working on electron microscopes he discovered that an object wave could be stored within an interferogram if a coherent background was added as a phase reference; he would later be awarded the Nobel prize in physics for his invention and development of the holographic method. Gabor coined the word holographic from the Greek *holos* and *graphe* meaning "whole drawing" because both the amplitude and phase of the object wave are recorded. As mentioned earlier, a requirement for holographic recording is that the object wave and reference wave must be coherent. Optical wave sources available at the time of Gabor's work had a coherence length of only a few mm, consequently application of his technique to optical waves could not produce quality images; that changed in 1960 with the invention of the laser. Visible light lasers have

coherence lengths of several metres, by arranging with a beam splitter (Fig. 1.4) the object and reference beams interfere to record a hologram on the material. By illuminating the hologram with monochromatic light with the same wavelength as the reference beam, a three-dimensional image of the object can be reconstructed.
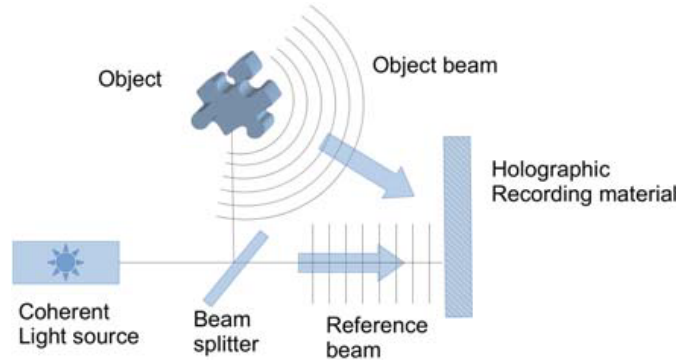
Fig. 1.4 Basic geometry of holographic recording [3]

A consequence of holographic recording is an alteration of the optical properties of the host material, specifically the RI. It is the spatial variation of the RI, rather than the physical structure, that stores the information in a hologram. The RI of a nanocomposite is calculated according to the Lorentz-Lorenz equation

$$\frac{n^2-1}{n^2+2} = \phi_m \frac{n_m^2-1}{n_m^2+2} + \phi_q \frac{n_p^2-1}{n_p^2+2} + \phi_b \frac{n_b^2-1}{n_b^2+2} \tag{1.3}$$

The variables ϕ_b, ϕ_m and ϕ_q refer to the fraction of the total volume made up by binder, monomer and polymer respectively; n_b, n_m and n_q are the respective RIs. The binder, sensitizing dye and electron donor are all non-reactive, the calculation can be simplified by treating all three as a single part whose fraction of the total volume is represented by ϕ_b; in general,

$$\phi_i = \frac{V_i(x,t)}{V_m(x,t) + V_q(x,t) + V_b(x,t)}.$$

Here x is the position within the grating space (see Fig. 1.3) and $t \geq 0$ is time. The binder is inert and immobile so its volume is constant through recording. The volume of monomer and polymer have both spatial-variation and temporal evolution and are denoted by

$$V_m(x,t) = \frac{m(x,t)}{\rho_m}, \qquad V_q(x,t) = \frac{q(x,t)}{\rho_q}, \qquad V_b(x,t) = \frac{b_0}{\rho_b}.$$

Expressions for $m(x,t)$ and $q(x,t)$ will be discussed further in Section 1.3; ρ_m, ρ_q and ρ_b are the respective mass densities, b_0 is the initial mass of binder. In a formula used in previous research at TU Dublin into photopolymer media [7] monomer accounts for 17% of the total volume of the photopolymer layer so that

$$b_0 = V_b(x, t=0)\rho_b = \frac{83}{17} V_m(x, t=0)\rho_b = \frac{83}{17}\frac{m_0}{\rho_m}\rho_b,$$

$$\frac{b_0}{m_0} = \frac{83}{17}\frac{\rho_b}{\rho_m},$$

where m_0 is the initial mass concentration of monomer. More generally, if the initial volume fractions are ϕ_b^0 and ϕ_m^0, then the mass ratios can be expressed as

$$\frac{b_0}{m_0} = \frac{\phi_b^0}{\phi_m^0}\frac{\rho_b}{\rho_m}. \tag{1.4}$$

The periodic nature of RI profile means it can be expressed as a Fourier series.

$$n(x, t) = N_0(t) + N_1(t)\cos(\frac{2\pi}{\Lambda}x) + N_2(t)\cos(\frac{4\pi}{\Lambda}x) + ...$$

If the spatial profile of the polymer is an accurate copy of the illumination pattern then the first harmonic will be a good approximation for the amplitude of the RI profile; hence, RI modulation is

$$\Delta n(t) \approx 2N_1(t) = \frac{4}{\Lambda}\int_0^\Lambda n(x, t)\cos\left(\frac{2\pi}{\Lambda}x\right)\,dx, \tag{1.5}$$

and the mean RI is modelled by the zero order harmonic

$$\overline{n}(t) = N_0(t) = \frac{1}{\Lambda}\int_0^\Lambda n(x, t)\,dx. \tag{1.6}$$

A scalar multiple of 2 is required because the first harmonic is the amplitude of the wave but Δn is the difference between max and min values. The advantage of measuring grating strength by the first harmonic is that the higher-order terms in the Fourier series are a good measure of how much the spatial profile of polymer deviates from the illumination pattern. The RI is approximated by the first two terms of the Fourier series, the higher order terms represent the deviation from the original interference pattern. Different recording conditions will result in different RI grating profiles so the distortion $\delta(t)$ is defined as the relative deviation between the RI and its Fourier representation

$$\delta(t) = \left\{\frac{1}{\Lambda}\int_0^\Lambda [n(x, t)]^2 dx\right\}^{-1}\frac{1}{\Lambda}\int_0^\Lambda \left\{n(x, t) - \left[N_0(t) + N_1(t)\cos\left(\frac{2\pi}{\Lambda}x\right)\right]\right\}^2\,dx \tag{1.7}$$

Polymer chains have a higher RI than monomer, for example at 532 nm the refractive indices of AA and polyacrylamide are 1.55 and 1.64 respectively [7] (at 633 nm 1.49 and 1.52 [21]), the spatial modulation of their concentration translates to a spatial modulation in the RI of the material. The RI modulation Δn is the difference between the RI of the polymer accumulated at the bright fringes of the interference pattern and the RI of the binder at the dark fringes; Δn can be measured indirectly

in the laboratory by measuring the diffraction efficiency (more in Section 1.5). Photopolymers may seem like a modern technology but knowledge and application of photosensitive materials dates back to antiquity. Bitumen of Judea, a naturally occurring asphalt soluble in spirits and oils, was known to harden when exposed to sunlight. Its earliest known use was in the process of mummification, later, Roman ship builders would exploit its properties for calking wooden ships. Today we understand this phenomenon is the result of photopolymerization and cross-linking. Research interest in photopolymers as recording media for holographic applications has increased substantially in recent years for many reasons: low production costs, self-processing, wide dynamic range and high sensitivity make them a suitable choice for holographic sensors [22, 23, 24]. As far as industrial applications go, the two properties of most interest are high contrast images and copying accuracy. The former requires minimal losses in optical power, the latter requires minimal distortion of the hologram. Maximizing RI modulation will yield a high diffraction efficiency needed to achieve high contrast imaging. To properly understand how the material properties and recording set up can be used to maximize Δn whilst simultaneously minimizing distortion of the hologram layer a mathematical model is needed to describe the photochemical reactions and mass transport of active components in the photopolymer system. In Section 1.3, a review on the history of mathematical models for the formation of holographic gratings in photopolymers will compare their predictions with observed behaviour.

1.3   Review of Earlier Models

A key assumption in modelling holographic recording in photopolymer media is that the photosensitive dye, the electron donor and the binder are inactive, any mention in the model of the binder will actually refer to all three. There are three dynamic parts in the photopolymer nanocomposite which are involved in photochemical reactions resulting in concentration gradients, two of which are sufficiently mobile that their concentration can be redistributed to minimize chemical potential. According to Fick's second law of diffusion, the general reaction-diffusion equation is

$$\frac{\partial \vec{u}}{\partial t} + \nabla J_{\vec{u}} = \vec{R}(\vec{r}, t)$$

where $\vec{J}_{\vec{u}}$ is the diffusion flux of vector $\vec{u}$ and $\vec{R}(\vec{r}, t)$ is the reaction. Diffusion flux is the measure of the flow of a substance per unit area per unit time, for some chemical with concentration u_i, its diffusion flux in some reference frame A is [25]

$$J_i^A = u_i(v_i - v_A)$$

where v_i is its local velocity and v_A is the reference frame velocity. For a nanocomposite of n parts the flow of any individual part can be eliminated by the definition of a reference frame. In this case it is the binder that is eliminated, in

other words, the mass transport of all other parts of the nanocomposite are relative to the binder [25].

Zhao & Mouroulis

In 1994 Zhao and Mouroulis published the first established mathematical model for describing holographic recording in photopolymer media [26]. The model uses a reaction-diffusion equation to describe the evolution of the concentration of monomer $m(x,t)$ and assumes that immobile polymer chains $q(x,t)$ grow at the local point of initiation. If $D(x,t)$ is the monomer diffusion coefficient and $F(x)$ the rate of polymerization then

$$\frac{\partial m}{\partial t} + \frac{\partial}{\partial x}\left[-D(x,t)\frac{\partial m}{\partial x}\right] = -F(x)m$$
$$\frac{\partial q}{\partial t} = F(x)m$$

The model assumes the intensity of the writing beams are sufficiently weak that there are no non-linear reactions so the rate of polymerization can be expressed as the product of a polymerization constant F_0 and a simple sinusoidal function $f(x)$ given by Eqn. 1.1

$$F(x) = F_0 f(x) = F_0\left[1 + V_0\cos\left(\vec{K}_g\cdot\vec{r}\right)\right] = F_0\left[1 + V_0\cos\left(\frac{2\pi}{\Lambda}x\right)\right] \qquad (1.8)$$

The authors initially assume that D should be non-uniform and a function of the polymer distribution and thus expressible as a Fourier series

$$D(x,t) = D_0(t) + D_1(t)\cos\left(\frac{2\pi x}{\Lambda}\right) + D_2(t)\cos\left(\frac{4\pi x}{\Lambda}\right) + D_3(t)\cos\left(\frac{6\pi x}{\Lambda}\right) + ...$$

It was then concluded in [26] that harmonics higher than the zero-order did not play a significant role and that a diffusion coefficient that decays throughout the recording would suffice

$$D(x,t) = D_m\exp\left[-\alpha F(x)t\right]; \qquad D_m, \alpha \in \mathbf{R}$$

In this equation, α represents a fixed rate of decay, [26] concluded this constant is sufficiently small that the monomer diffusion coefficient is uniform over both the grating period and the duration of recording, $D(x,t) = D_m$. Numerical simulations of the model can be performed by nondimensionalizing the variables as follows:

$$x = \Lambda\overline{x} \qquad t = t_0\overline{t} \qquad m = m_0\overline{m} \qquad q = m_0\overline{q}$$

The results of numerical simulations must be presentable in real-time values, hence $t_0 = 1$ s. The initial mass of monomer m_0 is a conserved quantity, since the Lorentz-Lorenz equation only needs the relative quantities the numerical simulations can be simplified by choosing $m_0 = 1$. Several theoretical models have been proposed to explain the poor performance of photopolymer gratings at high spatial frequencies [26, 27, 28, 29, 30, 31, 32]. Numerical simulations of the Zhao-Mouroulis model (Fig. 1.5) show that the model fails to predict the observed fall in Δn at high spatial frequencies.
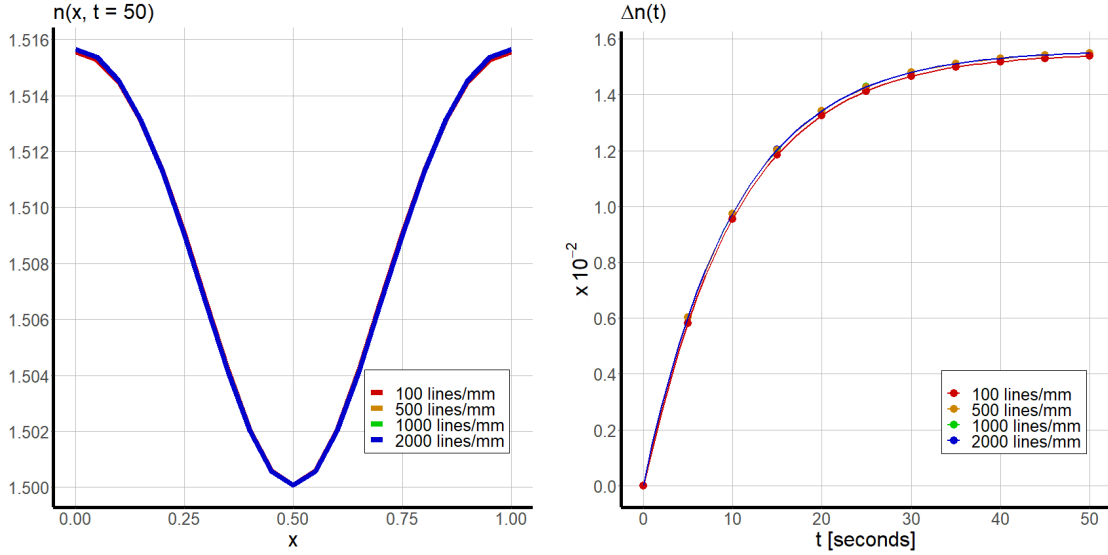
(a) RI at the end of holographic recording.   (b) RI modulation over 50 s recording.

Fig. 1.5 Numerical simulation of Zhao-Mouroulis model.

Non-Local Polymerization Driven Diffusion

The non-local polymerization driven diffusion model (NPDD) [33, 34] was the first to account for non-local effects, in other words that the material response at x can be influenced by what happens at x'. NPDD assumes that polymer chains can grow away from the point of initiation, spreading the polymer over the grating resulting in a flattened spatial profile. The mathematical description of the model uses a Gaussian function to account for random non-local growth of polymer chains

$$\frac{\partial m}{\partial t} = -\frac{\partial}{\partial x}\left[-D_m \frac{\partial m}{\partial x}\right] - \int_{-\infty}^{\infty}\int_0^t R(x, x'; t, t')F(x', t')m(x', t')dt'dx' \tag{1.9a}$$

$$\frac{\partial p}{\partial t} = \int_{-\infty}^{\infty}\int_0^t R(x, x'; t, t')F(x', t')m(x', t')dt'dx' \tag{1.9b}$$

The non-local response function $R(x, x'; t, t')$ describes the effect of the concentration of monomer at position x' and time t' on the rate of polymerization at x and time t. This function was further simplified by assuming that the time delay associated

with non-local growth of polymer chains was small compared to the characteristic time of polymerization and therefore instantaneous

$$R(x, x'; t, t') = R(x; x') = \frac{1}{\sqrt{2\pi\sigma}} \exp\left[\frac{-(x-x')^2}{2\sigma}\right]$$

The non-local response length $\sqrt{\sigma}$ represents the length over which significant non-local growth is observed. See that as $\sigma \to 0$ (Fig. 1.6) the spatial distribution of polymerization becomes more concentrated and consequently Eqn. 1.9 reverts to the Zhao-Mouroulis model.
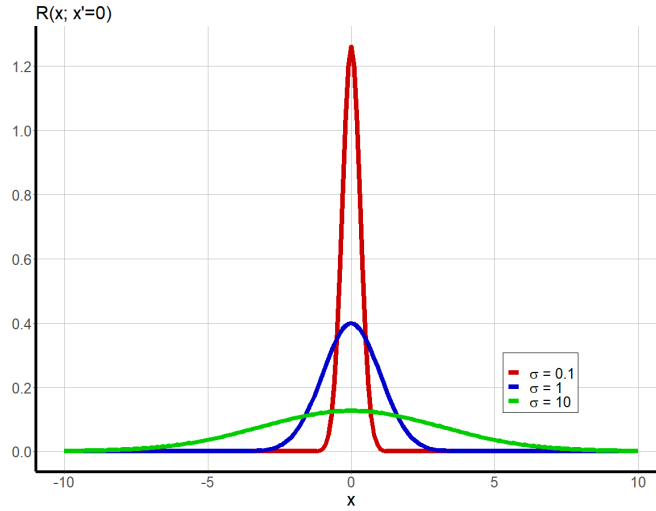
Fig. 1.6 The non-local response function; $x' = 0$ and $\sigma = 0.1, 1, 10$.

The NPDD model can predict the drop in performance at high spatial frequencies if one assumes there is some inverse proportional relationship between the grating period Λ and the non-local response σ. The model predicts that increasing the recording intensity should produce shorter chains which in turn should yield a better response, but high intensity recording conditions in experiment failed to yield an improvement. Consequently, a team of researchers from the Centre for Industrial and Engineering Optics (IEO) at TU Dublin preferred to pursue an alternative hypothesis: the two way diffusion model.

Two-Way Diffusion Model

In 2004, research by Naydenova [35] which aimed to find a diffusion coefficient for an AA monomer through real-time measurements of the diffraction efficiency over short exposures of a few tenths of a second concluded that there should be two mass transport processes taking place in the formation of holographic gratings in photopolymer media; one with a characteristic diffusion coefficient $D_1 = 1.6 \times 10^{-7}$ cm^2/s, and a second significantly slower in the opposite direction with a characteristic diffusion coefficient $D_2 = 6.35 \times 10^{-10}$ cm^2/s. The IEO team believed the first process must be the diffusion of the AA monomer and the second should be the diffusion of oligomer away from the bright fringes of illumination. The assumption that the diffusion of

oligomer has a significant role in the formation of holographic gratings forms the basis for the two way diffusion theory. Early mathematical descriptions for this theory [7] used two different variables to represent the concentration of mobile oligomer $p(x,t)$ and immobile polymer $q(x,t)$ and assumed that the formation of immobile polymer was the result of continuous propagation growth of polymer chains. The self-diffusion flux of the free monomer and oligomer is described in the model as the product of a constant diffusion constant and the respective concentration gradient.

$$\frac{\partial m}{\partial t} + \frac{\partial}{\partial x}\left[-D_m\frac{\partial m}{\partial x}\right] = -\Phi(t)F(x)m \tag{1.10a}$$

$$\frac{\partial p}{\partial t} + \frac{\partial}{\partial x}\left[-D_p\frac{\partial p}{\partial x}\right] = \Phi(t)F(x)m - \Phi(t)\Gamma mp \tag{1.10b}$$

$$\frac{\partial q}{\partial t} = \Phi(t)\Gamma mp \tag{1.10c}$$

The domain is $0 \leq x \leq \Lambda$ and $t \geq 0$, there is a zero-flux boundary condition and the initial conditions are: $m(x, t = 0) = m_0$, $p(x, t = 0) = 0$ and $q(x, t = 0) = 0$. The polymerization function can be expressed as the product of a polymerization constant and a sinusoidal function, $F(x) = F_0 f(x)$ as in the earlier Zhao-Mouroulis model. The rate of immobilization Γ is a constant; there is also a step function Φ introduced to facilitate the modelling of short exposures in which the exposure duration t_e isn't long enough for all available monomer to be consumed.

$$\Phi(t) = \begin{cases} 1 & t \leq t_e \\ 0 & t > t_e \end{cases}$$

The two way diffusion theory was further improved by changing the mechanism for immobilization. Originally, the hypothesis was that the formation of immobile polymer was due to the continuous propagation of oligomer, increasing in length until termination, as the polymer chains become increasingly massive their mobility drops off to zero. Propagation requires both monomer and oligomer present for the reaction to take place, as more and more monomer is consumed throughout holographic recording, the formation of immobile polymer should slow to halt, hence the reaction term Γmp. Further research with photopolymers concluded that the formation of immobile polymer is primarily the result of cross-linking of two or more oligomers [36, 37]; the mathematical description of the theory was updated to reflect this change [2, 38].

$$\frac{\partial m}{\partial t} + \frac{\partial}{\partial x}\left[-D_m \frac{\partial m}{\partial x}\right] = -\Phi(t)F(x)m \qquad (1.11\text{a})$$

$$\frac{\partial p}{\partial t} + \frac{\partial}{\partial x}\left[-D_p \frac{\partial p}{\partial x}\right] = \Phi(t)F(x)m - \Phi(t)\Gamma p^2 \qquad (1.11\text{b})$$

$$\frac{\partial q}{\partial t} = \Phi(t)\Gamma p^2 \qquad (1.11\text{c})$$

Several commercial photopolymer systems are available from Covestro, Du Pont, InPhase, IBM and Aprilis. With growing commercial interest in photopolymers the need for a quantitative measurement of the performance of different products available on the market also grew. The assessment of holographic recording media can be done by a characterization and comparison of their diffusion properties and rates of polymerization. The relationship between the rate of polymerization F_0 and the intensity of the recording beam I_0 emerged as one such metric with significant commercial and industrial interest. Early models assumed that the rate of initiation is equal to the rate of termination and the concentration of free-radicals stayed constant throughout polymerization and consequently inferred this relationship should be linear [39, 29, 26], later that F_0 is proportional to $\sqrt{I_0}$ [31]. IEO researchers believed it was better to determine the parameters of this relationship experimentally [40]. By recording the Raman spectra of two photopolymers, one made of AA and one a mixture of AA and BA, together with the spectra of their individual components, the peaks corresponding to relevant intermolecular interactions were identified, namely: 1629 cm^{-1} for the Carbon-Carbon double bond in BA, 1607 cm^{-1} for the Carbon-Carbon double bond in AA and 1281 cm^{-1} the bending mode of the CH vinyl bond on AA; by monitoring the intensities of these peaks throughout holographic recording for a range of different recording intensities, a relationship between the recording intensity and the characteristic time for decay of the intensity of the Raman peaks was established; defining the rate of polymerization as the reciprocal of this characteristic time the relationship is stated as

$$F_0 = k_p I_0^a \qquad (1.12)$$

where k_p is the polymerization constant and a is the illumination response constant, both are characteristics of the photopolymer system. The experiment found that the AA sample $k_p = 0.1 \text{ s}^{-1} \cdot \left[\text{mW/cm}^2\right]^{-a}$ and $a = 0.27$ and for the AA/BA sample $k_p = 0.114 \text{ s}^{-1} \cdot \left[\text{mW/cm}^2\right]^{-a}$ and $a = 0.32$ [40]. The value of Γ can be estimated by comparing numerical simulations of the model with experimental observations. The model equations require nondimensionalization, define independent variables as scalar products of a reference value

$$x = \Lambda \overline{x}, \qquad t = t_0 \overline{t}, \qquad m = m_0 \overline{m}, \qquad p = m_0 \overline{p}, \qquad q = m_0 \overline{q}.$$

The system of coupled partial differential equations (PDE) can be nondimensionalized by defining dimensionless constants

$$\alpha_m = \frac{D_m t_0}{\Lambda^2}, \qquad \alpha_p = \frac{D_p t_0}{\Lambda^2}, \qquad \beta = F_0 t_0, \qquad \gamma = \Gamma m_0 t_0.$$

Substituting the nondimensionalized parameters into Eqns. 1.11 (drop the line over the letters for the sake of legibility)

$$\frac{\partial m}{\partial t} = \alpha_m \frac{\partial^2 m}{\partial x^2} - \Phi(t)\beta f(x)m, \tag{1.13a}$$

$$\frac{\partial p}{\partial t} = \alpha_p \frac{\partial^2 p}{\partial x^2} + \Phi(t)\beta f(x)m - \Phi(t)\gamma p^2, \tag{1.13b}$$

$$\frac{\partial q}{\partial t} = \Phi(t)\gamma p^2 \tag{1.13c}$$

.

The domain of the non-dimensionalized PDEs is $0 \le x \le 1$ and $t \ge 0$. The initial conditions are now $m(x, t = 0) = 1$, $p(x, t = 0) = q(x, t = 0) = 0$. Numerical simulation of Eqn. 1.13 can be achieved with a Crank-Nicholson implicit finite difference scheme (see Appendix A)

$$\frac{m_i^{j+1} - m_i^j}{\Delta t} = \frac{\alpha_m}{2} \left[\frac{m_{i-1}^{j+1} - 2m_i^{j+1} + m_{i+1}^{j+1}}{\Delta x^2} + \frac{m_{i-1}^j - 2m_i^j + m_{i+1}^j}{\Delta x^2} \right] - \Phi^j \beta f_i \left(\frac{m_i^{j+1} + m_i^j}{2} \right), \tag{1.14a}$$

$$\frac{p_i^{j+1} - p_i^j}{\Delta t} = \frac{\alpha_p}{2} \left[\frac{p_{i-1}^{j+1} - 2p_i^{j+1} + p_{i+1}^{j+1}}{\Delta x^2} + \frac{p_{i-1}^j - 2p_i^j + p_{i+1}^j}{\Delta x^2} \right] +$$
$$\Phi^j \beta f_i \left(\frac{m_i^{j+1} + m_i^j}{2} \right) - \Phi^j \gamma p_i^j \left(\frac{p_i^{j+1} + p_i^j}{2} \right), \tag{1.14b}$$

$$\frac{q_i^{j+1} - q_i^j}{\Delta t} = \Phi^j \gamma p_i^j \left(\frac{p_i^{j+1} + p_i^j}{2} \right). \tag{1.14c}$$

Dividing m_i^j, p_i^j and q_i^j by ρ_m, ρ_p and ρ_q respectively will give us the volume of each part. Once one has expressions for the volume fraction of monomer, short polymer and cross-linked polymer one can use Eqn. 1.3 to get an expression for the RI as a function of x and t, $n(x, t) \approx n(x_i, t_j) = n_i^j$. The predicted RI modulation and distortion can be calculated using Simpsons' rule for numerical integration with Eqns. 1.5 and 1.7

$$\Delta n(t) \approx 2N_1(t_j) = 4\left[\frac{\Delta x}{3}\left[n_0^j\cos(2\pi x_0) + 4n_1^j\cos(2\pi x_1) + 2n_2^j\cos(2\pi x_2) + \ldots\right.\right.$$
$$\left.\left. +2n_{N-2}^j\cos(2\pi x_{N-2}) + 4n_{N-1}^j\cos(2\pi x_{N-1}) + n_N^j\cos(2\pi x_N)\right]\right]$$

$$N_0(t_j) \approx \frac{\Delta x}{3}\left[n_0^j + 4n_1^j + 2n_2^j + \ldots + 2n_{N-2}^j + 4n_{N-1}^j + n_N^j\right]$$

Numerical simulations allow researchers to investigate how the photopolymer material properties and recording setup can be used to optimize the holographic grating for specific applications. Numerical results of the time evolution of RI modulation and distortion for a range of different spatial frequencies are presented in Fig. 1.7.

(a) RI modulation (b) Distortion
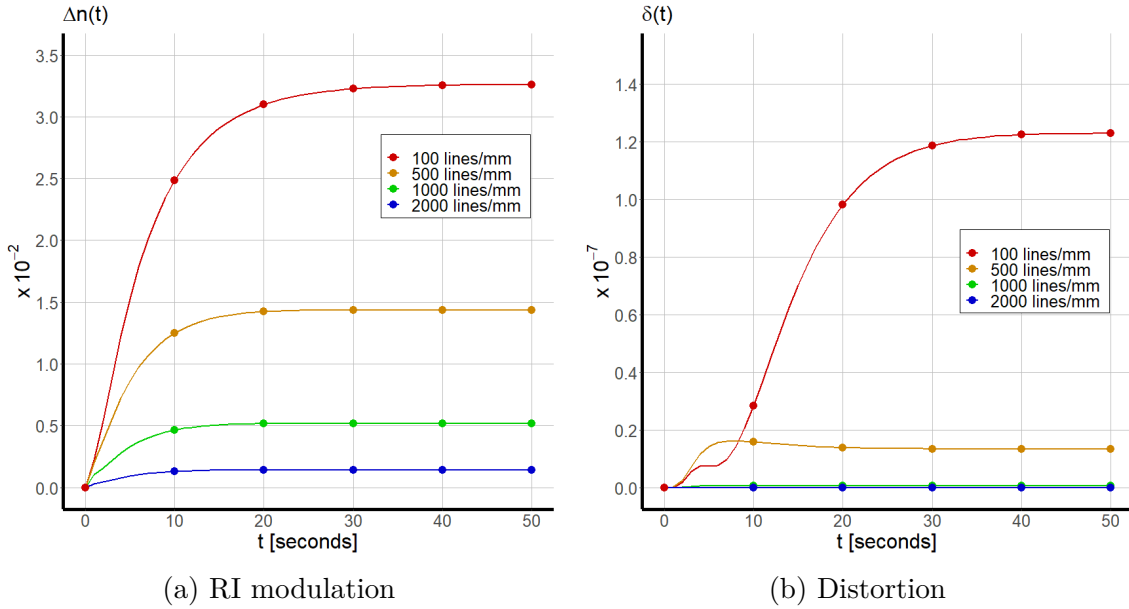
Fig. 1.7 Two way diffusion model: the effect of the spatial frequency of recording.

For a relatively low spatial frequency of 100 lines/mm the distance between a bright fringe and an adjacent dark fringe is 5 μm; by increasing the spatial frequency to 2000 lines/mm that distance is reduced to just 0.25 μm. The mobile oligomer has a significantly shorter distance to cover so the grating profile of polymer growth is more uniform and spatial modulation of the RI is very small. Previous modelling strategies [2, 38, 33, 26] have found that one must consider the ratio of nondimensional constants, particularly monomer diffusion α_m and polymerization β to understand the response to changes in spatial frequency. The nondimensionalized diffusion to polymerization ratio is calculated as

$$\kappa = \frac{\alpha_m}{\beta} = \frac{D_m}{F_0\Lambda^2}. \tag{1.15}$$

For $\kappa >> 1$ diffusion processes are dominant; monomer can move from dark to bright fringes in less time and can thus be replaced as quickly as they're consumed, however, oligomer can also move from bright to dark fringes in less time resulting in greater distortion and greater formation of polymers at dark fringe positions. Whereas if $\kappa << 1$ polymerization is dominant; oligomer takes much longer to get from bright to a dark fringe so there should be very little distortion but monomer will also take longer to go from a dark to bright fringe, consequently there is less polymer growth at the bright fringes.

(a) RI modulation   (b) Distortion
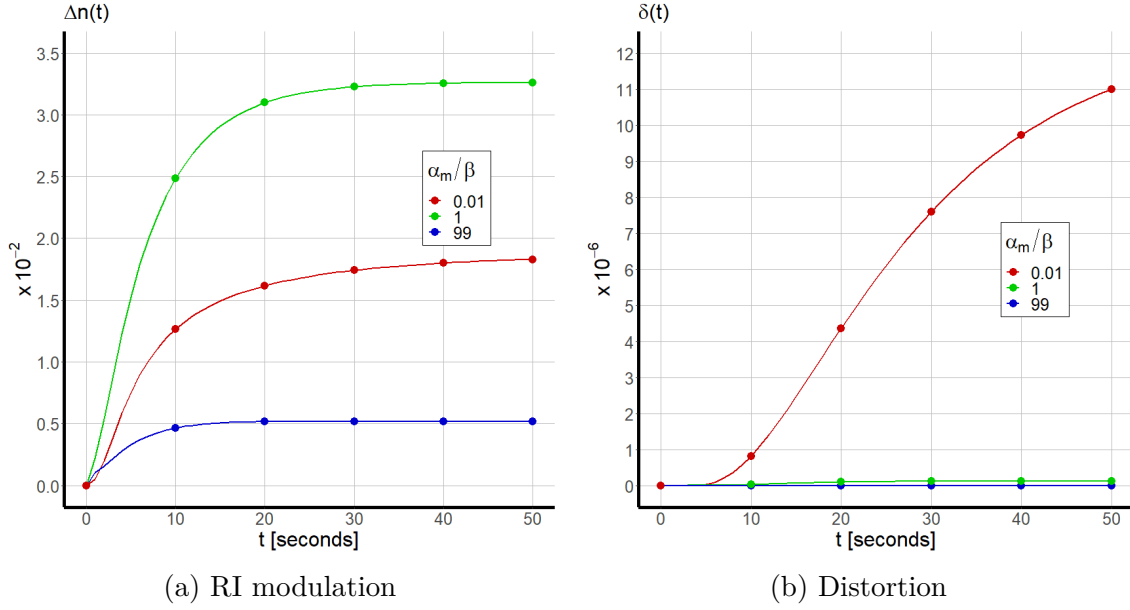
Fig. 1.8 Two way diffusion model: non-dimensional diffusion to polymerization ratio.

The non-dimensional diffusion to polymerization ratio κ can be varied by changing either monomer diffusivity, the spatial frequency or the recording intensity; the method by which the ratio is changed makes little difference with regards to the final properties of the resulting holographic grating. In Fig. 1.8 κ is varied by using three different spatial frequencies 10 lines/mm ($\kappa = 0.01$), 100 lines/mm ($\kappa = 1$) and 1000 lines/mm ($\kappa = 100$); Δn is maximized for $\kappa \approx 1$, whereby neither polymerization nor diffusion dominate over the other. For a polymerization dominant set up, $\kappa << 1$, the rate at which monomer is consumed at the bright fringes is much faster than the rate at which it can be replenished, hence a much longer exposure duration is required to polymerize all the available monomer. This is highly inefficient, not only will a longer exposure consume more energy but much of that energy cannot be utilized as the monomer cannot diffuse to the bright fringes fast enough to participate in the photochemical reaction. A further consequence $\kappa << 1$, the high concentration of monomer remaining near the dark fringes means an increase in higher order harmonics which in real terms means a deterioration of copying accuracy. For $\kappa >> 1$, the monomer is replenished as fast as it can be consumed at the bright fringe but oligomer is also more mobile leading to greater growth of immobile polymer at dark fringe positions and a smaller Δn.

(a) RI modulation   (b) Distortion
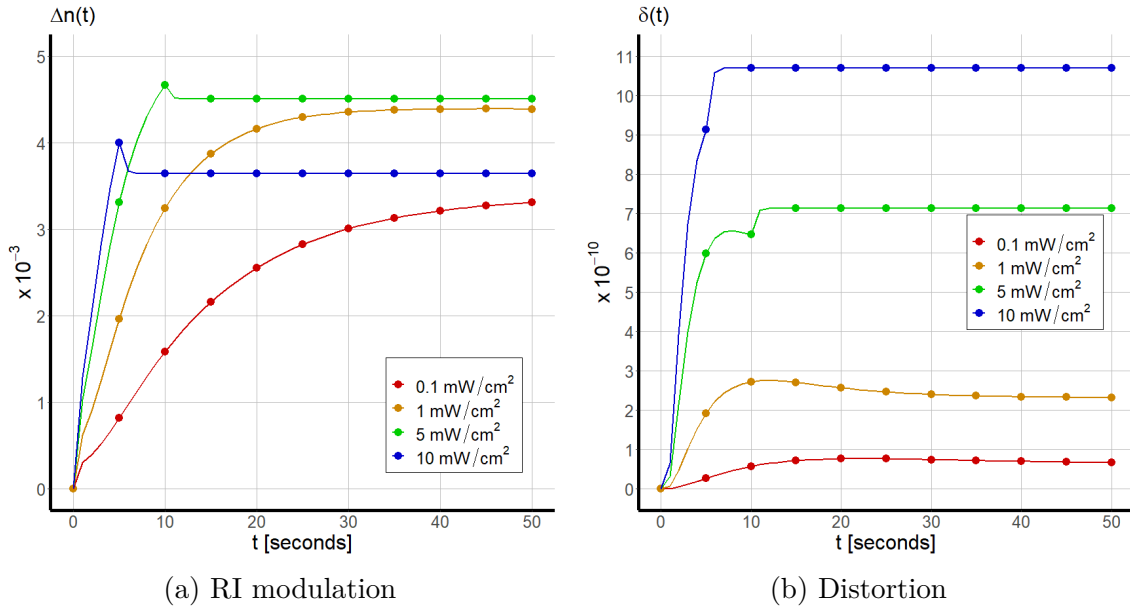
Fig. 1.9 The change in RI modulation and distortion predicted by the two way diffusion model for recording intensities of 0.1, 1, 5 and 10 mW/cm$_2$.

The results of a numerical simulation of the two way diffusion model for four different recording intensities 0.1, 1, 5 and 10 mW/cm^2 are presented in Fig. 1.9. The exposure duration is 50 s, 10 s and 5 s respectively so that there is a constant energy density of 50 mJ/cm^2. The two way diffusion model predicts that for a fixed spatial frequency of 1000 lines/mm there is an optimal recording intensity between 1 and 5 mW/cm^2. At 0.1 mW/cm^2 the rate of polymerization is too weak and very little polymer is formed within the 50 s recording time. At 10 mW/cm^2 the rate of polymerization is much higher resulting in a fast growth in Δn, but after 5 s recording ends and there is a substantial quantity of polymer formed near the point of initiation, diffusion of the mobile short polymer results in a drop in Δn before settling on a new equilibrium. A high intensity means more photons per unit area and hence more photoinitiation at the bright fringes and more polymer growth, but the monomer is consumed in the photochemical reactions faster than it can be restocked, consequently there is a substantial portion of the initial monomer unpolymerized after the illumination is switched off so Δn can't reach the large values in the lower intensity recordings. Furthermore, when the illumination is switched off there is still some oligomer remaining, both the unpolymerized monomer and oligomer diffuse and Δn will drop off until their grating profiles become uniform.

(a) RI modulation
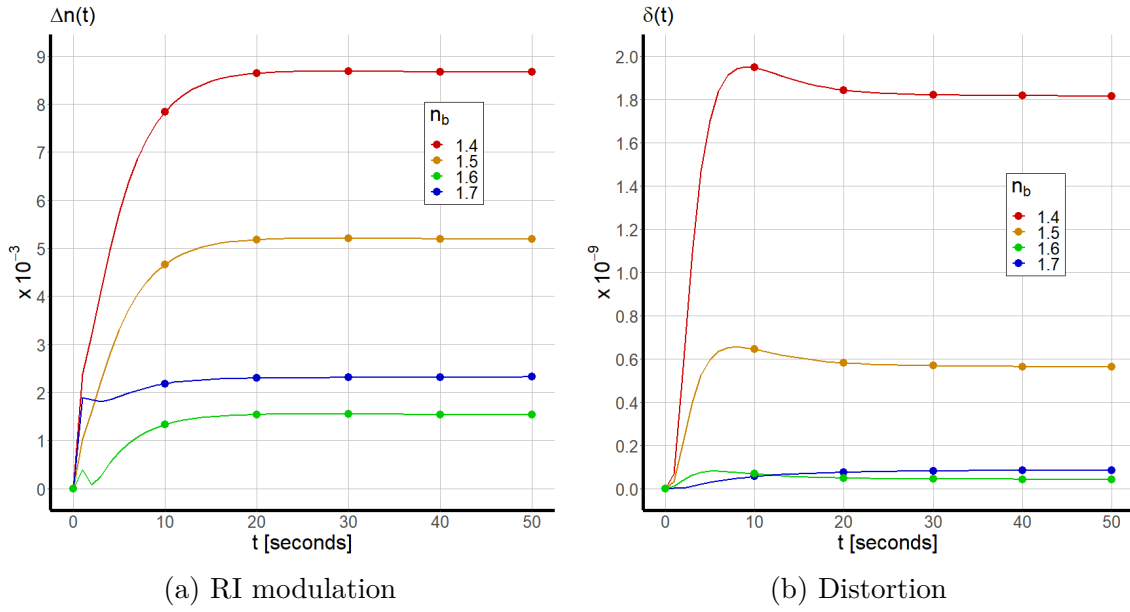
(b) Distortion

Fig. 1.10 The change in RI modulation and distortion predicted by the two way diffusion model for a binder RI of 1.4, 1.5, 1.6 and 1.7.

The RI modulation is the difference between the RI of material at the bright fringe and material at the dark fringe. Once holographic recording is complete, the bright fringe areas will be composed of a mixture of polymer chains and binder while the dark fringe areas are just binder. Model predictions show the choice of a binder with a RI either much higher, or much lower, than the RI of the polymer (polyacrylamide $n_q = 1.64$) will achieve the largest possible Δn.

Figs. 1.7, 1.8, 1.9 and 1.10 demonstrate that the most recent version of the two way diffusion model (Eqn. 1.11) is capable of reproducing experimental observations regarding the performance of holographic gratings recorded in AA/BA photopolymer media, most notably the poor performance at high spatial frequencies and deterioration of copying accuracy in weak diffusion-strong polymerization recording conditions. Both these caveats must be addressed if photopolymer based holographic gratings are to be a feasible technology; one possible solution is the introduction of inorganic nanoparticles.

1.4 Hybrid Photonic Structures

According to Kogelnik's coupled wave theory, a large RI modulation allows for greater diffraction efficiency without needing to fabricate thicker layers and hence is advantageous for many holographic applications. The numerical results of the two way diffusion model presented in the previous section (Figs. 1.7, 1.8, 1.9 and 1.10) show that Δn can be maximized by the use of a monomer, cross-linking monomer and binder with the ideal mechanical and optical properties, and by controlling the recording conditions. An alternative approach is the incorporation

of small amounts of inorganic nanoparticles into the photopolymer mixture [41].
The diffusion of inert nanoparticles from bright to dark fringes has been observed
[41] in holographic recording [42]; spatial redistribution of nanoparticles with a
RI significantly different to that of the polymer will make a positive contribution
to Δn. Mechanical properties of inorganic nanoparticles can provide additional
benefits for the photopolymer; high mass density and high elastic modulus give
increased mechanical stability, increased sensitivity and reduce photopolymerization-
induced shrinkage. Holographic recording in photopolymers doped with inorganic
nanoparticles has successfully achieved large dynamic range and reduced shrinkage
using metal-oxides such as ZrO_2 [41], TiO_2 [43, 44], SiO_2 [45, 46]; pure elemental
metals such as gold [47, 48] and silver [49, 50, 51, 52]; quantum dot nanoparticles
such as CdSe [53] and ZrO [54] have also proven successful. Researchers at IEO have
done extensive work with zeolites [55, 56, 57, 58, 6], porous inorganic nanoparticles
formed by a network of interconnecting silica tetrahedra. A variety of different
zeolite-nanostructures can be fabricated in the laboratory (Fig. 1.11).

(a) MFI   (b) AEI   (c) BEA
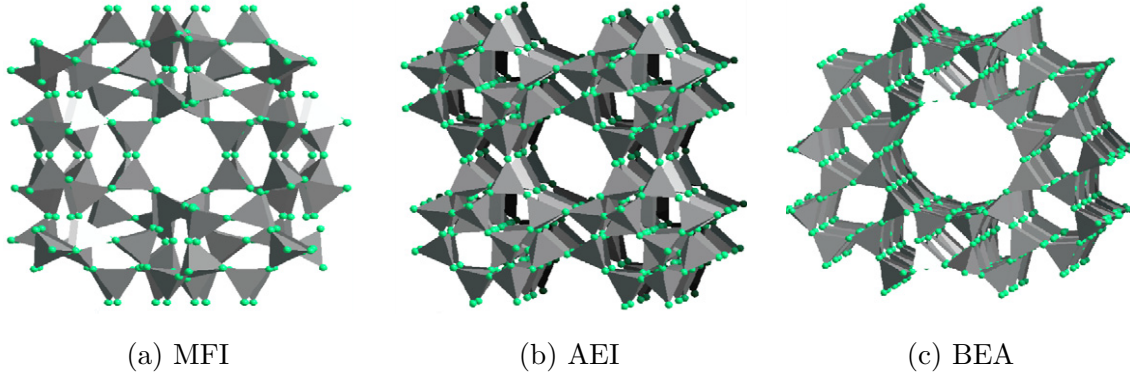
Fig. 1.11 Zeolite frameworks

	ρ	n	Pore size
MFI	1.74 g/cm^3	1.366	5.1×5.5 Å
BEA	2.24 g/cm^3	1.456	7.6×6.4 Å
AEI	2.42 g/cm^3	1.414	3.8×3.8 Å

Table 1.1 Mass density, RI and pore size of three different zeolite frameworks. [6]

Different frameworks have different physical properties (Table. 1.1); particle
size, pore size, mass density, elastic modulus, hydrophilic/hydrophobic, RI, etc.
can be fine tuned to optimize photopolymer functionality. Researchers from IEO
investigated holographic recording in AA based photopolymers doped with three
different nanoparticles: beta, AIPO-18 and silicate-1 nanozeolites [6]. The beta
nanoparticles have the BEA-structure and a diameter 40 nm, the pore size is slightly
larger than the diameter of the AA monomer molecules (approximately 5×4 Å)
allowing for migration of monomer into the zeolite pores in the preparation stage;
the AIPO-18 nanoparticles have the AEI-structure and diameter 180 nm, AIPO-18

are hydrophilic so their pores are occupied with water molecules; consequently, the addition of beta or AIPO-18 nanozeolites to the nanocomposite yield almost no improvement to Δn. The silicate-1 nanoparticles have the MFI-structure and a diameter of 30 nm, the smaller pore size and hydrophobic nature ensured the pores remained vacant throughout preparation and recording, the redistribution of silicate-1 nanoparticles contributed to an improvement in Δn; the inclusion of 5 wt.% silicate-1 nanozeolites has shown to increase diffraction efficiency by 40% [59, 4]. By increasing the quantity of inorganic nanoparticles in the photopolymer mixture, a greater concentration is expected to accumulate at the dark fringes in holographic recording and subsequently yield a greater improvement in Δn. Cody et al. investigated the effects of increasing the concentration of beta nanozeolites in a Diacetone-acrylamide-based photopolymer [55], the results were very different from expectations (Fig. 1.12a).

(a) Real-time Δn [55]

(b) Volume shrinkage [59]
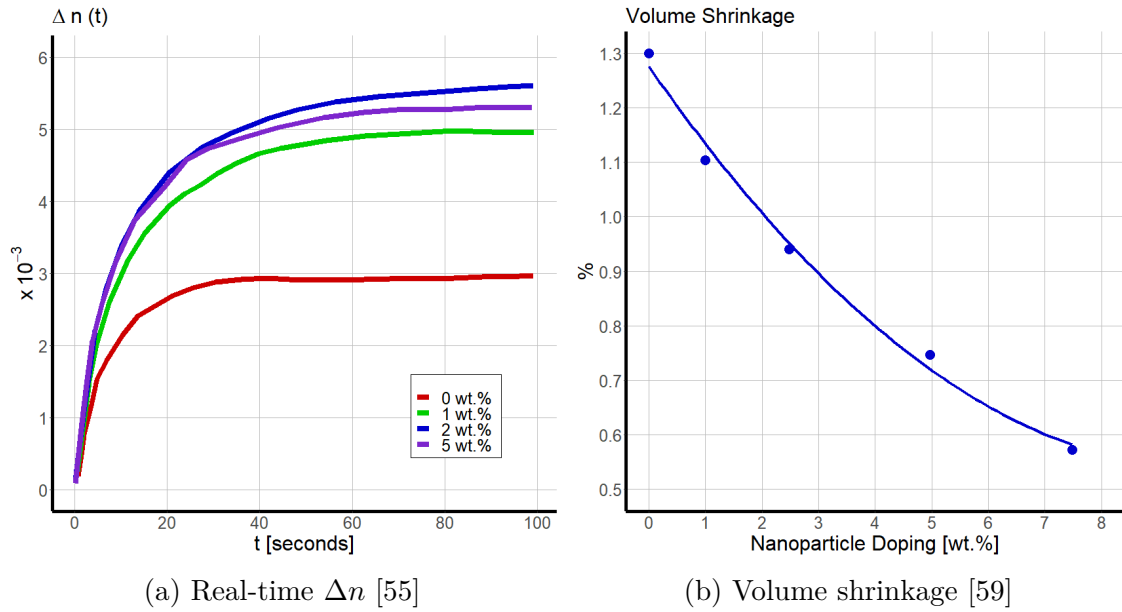
Fig. 1.12 Observed results of holographic gratings recorded in zeolite doped photopolymers; RI modulation and volume shrinkage dependence on nanoparticle concentration.

Increasing concentration from 0 wt.% to 1 wt.% yields a large increase in Δn (almost 67%), but increasing the nanodopant concentration to 2 wt.% yields a much smaller improvement in Δn and increasing to 5 wt.% resulted in a disimprovement in Δn. In addition to the contribution to the RI modulation, the advantage of zeolite nanoparticles in photopolymers is the reduction in photopolymerization-induced volume shrinkage. Holographic recording in AA based photopolymers doped with silicate-1 nanozeolites has been demonstrated in [59], the authors recorded the slant angle before ϕ_0 and after ϕ_1 holographic recording and calculated the fractional change in grating thickness as[4]

$$\frac{\Delta T}{T} = 1 - \frac{\tan \phi_0}{\tan \phi_1}. \tag{1.16}$$

Material shrinkage dropped from 1.32% in the undoped sample to 0.57% in samples doped 7.5% with nanozeolites (Fig. 1.12b). A key measurement of success for this research is that any mathematical model developed must be capable of predicting these results.

1.5   Holographic Sensors

The original wavefront recorded in a hologram can be reconstructed via illumination with a reference wave. Both the hologram and the reference beam must be in the same condition as in the recording stage to construct a perfect image. Any deviation in the condition of the hologram or in the reference beam will result in a change in properties of the reconstructed wavefront. Holographic sensors exploit this phenomenon by calibrating small dimensional changes or changes in the optical properties against small changes in the hologram's local environment. Changes to the environment detectable by holographic sensors include the concentration of a target analyte in a solution to which the hologram is exposed; pressure; temperature; humidity [60, 23, 61]. Detectable changes to the hologram include changes in the angular response, the wavelength response and the diffraction efficiency.

Kogelnik's coupled wave theory accurately describes the optical properties of volume phase holograms [11]; for a holographic grating with a thickness T, recorded using writing beams with wavelength λ_r at Bragg angle θ_B, the diffraction efficiency η (the ratio of diffracted beam power to incident beam power) is a function of the RI modulation Δn

$$\eta = \sin^2\left(\frac{\pi \Delta n T}{\lambda_r \cos\theta_B}\right)$$

In the reconstruction stage, the holographic grating is exposed to an environment that differs from the recording environment resulting in small changes to either Δn or T (swelling or shrinkage). In the case of reflection holograms λ_r or θ_B can change in response to environmental changes.

1.6   Outline of Thesis

The first primary objective of this research is to augment and extend the established mathematical model [2, 38] for holographic recording in photopolymer media to novel hybrid materials, in particular, AA/PVA based photopolymers doped with zeolite nanoparticles. A successful model will be capable of predicting the redistribution of nanoparticles and quantifying how the physical and optical properties of the nanocomposite are influenced by the physical and optical properties of the nanoparticles throughout holographic recording. In Chapter 2 a new mathematical model will be introduced with a description and justification of the new terms incorporated into the model. A brief description of the numerical scheme used for running simulations

of the model will be provided in Appendix A. Results of numerical simulations will be presented with a qualitative comparison with observed behaviour. A secondary objective is to extend the new model to two spatial variables with the view to study and gain a better understanding of polymerization-induced shrinkage, modelling how the sample thickness effects the holographic recording, how the optical properties of the hologram change with depth within the sample and predicting the behaviour of a holographic recording of a two-dimensional illumination pattern. Chapter 3 will describe how the new model can be extended to a second spatial dimension, the required changes to the numerical scheme, and the results of numerical simulation of the extended model. The second primary objective is to model the time response of holographic sensors based on the diffusion dynamics of a specified target analyte and structure of the hologram sensor operating in either reflection or transmission model. Chapter 4 will focus on modelling the dynamics of a holographic sensor, specifically how the optical properties and physical dimensions will change due to the diffusion of a target analyte solution through a theoretically modelled holographic grating. Chapter 5 will cover possibilities for future research building upon the findings of this project. A summary of the conclusions of this research project in its entirety will be documented in Chapter 6.

Chapter 2

Unslanted Holographic Gratings in Hybrid Nanocomposites

The two way diffusion model described in Chapter 1 has proved successful in predicting the non-local growth of polymer chains and the drop in Δn at high spatial frequencies. However, the model needs to be augmented to describe the experimentally observed redistribution of inorganic nanoparticles [42, 6] during holographic recording in hybrid photopolymer media. In this chapter we address the first of our research questions: *can a mathematical model predict the redistribution of inorganic nanoparticles in a hybrid photopolymer during holographic recording?* To that end we will outline how the two way diffusion model can be augmented for the purpose of modelling mass transport of inorganic nanoparticles and the formation of unslanted holographic gratings in hybrid nanocomposites along with the required changes to the numerical scheme. By *unslanted* we mean that the planes in which the bright fringes of the interference pattern are found are perpendicular to the surface of the recording medium, hence only one spatial variable is required. The model's performance will be assessed via a comparison with experimental observations. We will end the chapter by investigating, via numerical simulation of the model, how holographic gratings can be optimized with the proper selection of recording conditions and nanocomposite sample preparation for holographic applications.

2.1 Optical Losses in Holographic Recording

A sufficiently large difference between the RI of inorganic nanoparticles and the host photopolymer matrix can result in significant photon scattering. It has been experimentally observed [6] that optical losses in hybrid photopolymers due to scattering increased with the concentration of nanoparticles; scattering losses also varied with the type of nanoparticle framework, BEA zeolites displayed greater optical losses compared with MFI or AEI type structures. This research proposes that optical losses in holographic recording can be modelled by a reduction in the visibility of the sinusoidal illumination pattern. If Λ is the grating period, x is the

distance across the grating ($0 \leq x \leq \Lambda$) and $t \geq 0$ is the time since the start of holographic recording, the polymerization function derived and used in Chapter 1 (Eqn. 1.8) can be modified to model optical losses

$$F(x,t) = F_0 f(x,t) = F_0 \left[1 + V e^{-\xi z} \cos \left(\frac{2\pi}{\Lambda} x \right) \right], \tag{2.1}$$

where ξ is the scattering coefficient which governs the rate at which the visibility V will decay with the concentration of nanoparticles $z(x,t)$. Experimental studies [6, 62, 58] of holographic recording in hybrid photopolymer media have found that scattering is a function of nanoparticle size and the difference between the nanoparticle RI and the RI of the surrounding material. The rate of polymerization F_0 in photopolymers has been estimated as

$$F_0 = k_p I_0^a,$$

where I_0 is the recording intensity measured in mW/cm^2; $k_p = 0.1 \text{s}^{-1} \cdot [\text{mW/cm}^2]^{-a}$ and $a = 0.3$ for an AA/PVA photopolymer [40].

2.2   RI Modulation

Predictions for the change in RI modulation in response to increased doping with inorganic nanoparticles will be presented. As discussed in Chapter 1, the volume of monomer, short polymer, cross-linked polymer and inorganic nanoparticles can be calculated by dividing solutions to the model PDEs by the appropriate densities. If the RI and volume fraction of each part of the nanocomposite is known (see Table. 2.1) then solving the Lorentz-Lorenz equation,

$$\frac{n^2 - 1}{n^2 + 2} = \phi_m \frac{n_m^2 - 1}{n_m^2 + 2} + \phi_p \frac{n_p^2 - 1}{n_p^2 + 2} + \phi_q \frac{n_q^2 - 1}{n_q^2 + 2} + \phi_z \frac{n_z^2 - 1}{n_z^2 + 2} + \phi_b \frac{n_b^2 - 1}{n_b^2 + 2}, \tag{2.2}$$

will give the RI of the nanocomposite as a function of the position within the grating period and time, $n(x,t)$. The RI modulation at time t can be calculated as the difference between the maximum and minimum values at t. However, because the illumination pattern is a sinusoidal function the nanocomposite RI can be represented by a Fourier expansion series, the first order harmonic in this series multiplied by two is an approximation for the RI modulation (Eqn. 1.5). The higher order harmonics in the Fourier series are a measure of distortion (Eqn. 1.7), in other words, the extent to which the holographic grating deviates from the illumination pattern.

	ρ [g/cm^3]	n
Monomer (AA)	1.13	1.55
Short Polymer (AA)	1.13	1.55
Cross-linked Polymer (AA)	1.30	1.64
Binder (PVA)	1.19	1.5
MFI-Zeolite	1.74	1.366
BEA-Zeolite	2.24	1.456
AEI-Zeolite	2.42	1.414

Table 2.1 Density and RI [6, 7, 8].

2.3   Fraction of Redistribution

Earlier studies in holographic recording in photopolymers doped with zeolite nanoparticles have calculated the fraction of nanoparticles redistributed in recording using Raman spectroscopy [6] with different nanostructures displaying different redistribution; 38% of MFI zeolites, 31% of AEI zeolites and 40% for BEA (other studies with BEA zeolites have also reported 40% [22] and 43% [55]). An investigation [22] into holographic recording in hybrid photopolymers recorded 200 lines/mm pattern with intensity 5 mW/cm^2 for a 120 s duration onto an AA/PVA photopolymer doped 5 wt.% with BEA nanozeolites and reported an estimated 40% redistribution. There are alternative experimental techniques for measuring the fraction of inorganic nanoparticles redistributed in holographic recording. For example, if v_z is the volume fraction of nanoparticles in the nanoparticle-rich area of the grating then [63]

$$v_z = \frac{\pi \Delta n}{2(n_q - n_z)}.$$

The fraction of redistribution is v_z divided by the initial volume fraction of nanoparticles. However, Eqn. 2.3 is a good approximation only for gratings that approximate a square wave, in reality the recording wavefront produced by the holographic recording set-up is sinusoidal. Figure 2.1 shows a simplified view of the redistribution of inorganic nanoparticles.
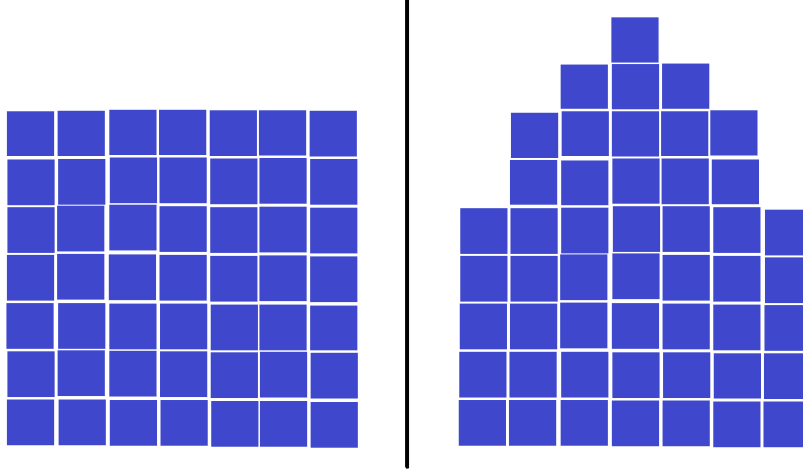
Fig. 2.1 Before exposure (left) and after holographic recording is complete (right).

On the left, before exposure 0% of nanoparticles are redistributed. On the right, when holographic recording is finished 4 of the 49 particles have moved from the outermost part (the bright fringe) and redistributed to the inside (the dark fringe), in this case redistribution is $4/49 \approx 8.2\%$. The fraction of redistribution can be calculated using solutions to the model PDEs. Define

$$G(x,t) = \left\{ \begin{array}{ll} z(x,t) - z(x,0) & \text{if } z(x,t) > z(x,0) \\ \\ 0 & \text{if } z(x,t) \leq z(x,0) \end{array} \right\},$$

$$\text{Fraction of redistribution} = \frac{\frac{1}{\Lambda} \int_0^\Lambda G(x,t) \ dx}{\frac{1}{\Lambda} \int_0^\Lambda z(x,0) \ dx}. \tag{2.3}$$
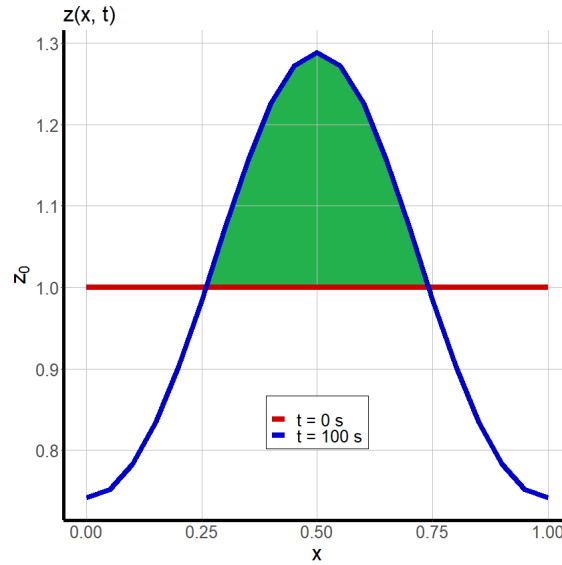
Fig. 2.2 Grating profile of mass concentration of inorganic nanoparticles at $t = 0$ and $t = 100$.

By nondimensionalizing the $z(x,t)$ and x the denominator of Eqn. 2.3 will equal one. The area of the green region in Figure 2.2 is an approximation of the fraction of inorganic nanoparticles redistributed. The fraction of nanoparticles redistributed in holographic recording will depend upon a combination of different recording conditions, namely, the spatial frequency; monomer mobility; recording intensity; short polymer mobility; and the rate of cross-linking.

2.4   Modelling Diffusion

A key assumption of this model is that inorganic nanoparticles added to the photopolymer are inert. Because there are no chemical reactions involved it is intuitive that the grating profile of inorganic nanoparticles should remain uniform throughout holographic recording, however, there is evidence [42, 47, 49, 50, 51] that nanoparticles are redistributed in the recording. If there are no photochemical reactions then the mass transport of inorganic nanoparticles can be described by a one-dimensional diffusion equation

$$\frac{\partial z}{\partial t} + \frac{\partial J_z}{\partial x} = 0, \tag{2.4}$$

where J_z is the diffusion flux for inorganic nanoparticles in holographic recording. The same zero-flux boundary conditions applied to the monomer and short polymer in the two way diffusion model for holographic grating formation in pure-organic photopolymer media will also apply to the new equation,

$$\frac{\partial z(0,t)}{\partial x} = \frac{\partial z(\Lambda, t)}{\partial x} = 0. \tag{2.5}$$

By the end of the preparation stage for the hybrid photopolymer material the inorganic nanoparticles have a uniform distribution, hence the initial conditions for the new diffusion equation are,

$$z(x,0) = z_0. \tag{2.6}$$

The initial mass concentration of inorganic nanoparticles z_0 and binder b_0 will need to be expressed in terms of the initial mass of monomer m_0. If the inorganic nanoparticles are non-porous, or if the pores of the nanoparticles are sufficiently small that monomer cannot penetrate and migrate inside, then the calculation is straightforward. The amount of each part is usually reported in the literature by its weight percentage (wt.%). If Z is the weight percentage of inorganic nanoparticles in the hybrid material then,

$$Z = \frac{z_0}{z_0 + m_0 + b_0},$$

this equation can be rearranged,

$$\frac{z_0}{m_0} = \left(\frac{Z}{1-Z}\right)\left(1 + \frac{b_0}{m_0}\right). \tag{2.7}$$

For example, from the experimental set-up described in [7] $b_0/m_0 = 5.14$; if this photopolymer is doped 5 wt.% with MFI nanozeolites, Eqn. 2.7 yields $z_0/m_0 = 0.32$. The initial mass of monomer in the sample m_0 as a whole can be calculated with sufficient information about the experimental set-up: the density of the molecule(s) to use as a monomer, concentration in the mixture, the surface area of the sample and the initial thickness of the photopolymer layer. However, this theoretical model restricts its predictions to the spatial-temporal evolution of the physical parameters across a single diffraction grating, consequently, m_0 is the initial mass concentration of monomer within the volume of a single grating and establishing a value for it requires additional information: the spatial frequency of the illumination pattern; the number of diffraction gratings formed; whether the pattern projected over the length or width of the sample; and the distance over which the pattern is projected. But the predictions of a practical theoretical model should be isotropic and independent of the physical size of the sample. It suits the purposes of this theoretical study to treat the initial mass of monomer as a reference value, i.e. all other mass concentrations can be expressed as relative to m_0. If porous nanoparticles with a pore diameter larger than the size of the monomer molecule are added instead then some fraction of the monomer will migrate into the nanoparticle. If monomer capture is possible then it should take place during the preparation stage, the timescale of which is 12-24 hours, rather than the recording stage which typically takes place over a period of 10-300 seconds so the process of monomer capture by porous nanoparticles will be ignored in the consideration of any model for holographic recording. However, there are still some changes to the initial conditions which should be taken into account if there is migration of monomer molecules to the nanoparticle interior, namely the ratio of binder to monomer mass, the ratio of nanoparticle to monomer mass and the RI of monomer-occupied porous nanoparticles should differ to the RI of vacant porous nanoparticles. If W is the fraction of the available monomer that migrates into the porous nanoparticle one should expect that the adjusted initial relative mass of binder and nanoparticles will be

$$\frac{b_0^*}{m_0^*} = \frac{1}{1-W}\frac{b_0}{m_0} \qquad \text{and} \qquad \frac{z_0^*}{m_0^*} = \frac{1}{1-W}\left(\frac{Z}{1-Z}\right)\left(1 + \frac{b_0^*}{m_0^*}\right).$$

The density of occupied and vacant porous nanoparticles will be different but the volume will not change. The nanocomposite RI is a function of the volume fraction of its individual parts, hence the change in density is not important to this analysis.

Perhaps the most straight forward approach for modelling diffusion flux of inorganic nanoparticles is to assume that mass transport is driven by a counter

action to the mass transport of the monomer and mobile short polymer, this is called counter-diffusion and the nanoparticle diffusion flux can be expressed as,

$$J_z + J_m + J_p = 0,$$

where J_m and J_p are the diffusion fluxes for monomer and short polymer respectively. There are two major caveats with this idea: 1) there are no self diffusion terms in J_z; 2) as mass transport of monomer and short polymer continues the nanoparticles will continue to be pushed away from the bright fringes, when the mass concentration of nanoparticles at $x = 0, 1$ reaches zero the redistribution does not stop unless the diffusion of monomer and short polymer hasn't also stopped, consequently, the predicted nanoparticle mass drops into negative values. As long as J_m or J_p is non-zero, J_z must also be non-zero. Clearly, counter-diffusion is not an acceptable hypothesis for the redistribution of nanoparticles. An alternative hypothesis for the redistribution of inorganic nanoparticles is cross-diffusion, i.e. mass transport induced by the concentration gradient of a different component of the same system [25]. The mass concentration of nanoparticles $z(x, t)$ will be modelled by the one dimensional diffusion equation with the initial and boundary condition defined earlier in this section. In an earlier publication [64], we proposed that the diffusion flux is equal to the sum of self-diffusion and cross-diffusion induced by the concentration gradient of cross-linked polymer chains.

$$J_z = -D_{zz}\frac{\partial z}{\partial x} - D_{qz}z\frac{\partial q}{\partial x}, \tag{2.8}$$

where D_{zz} is the self-diffusion coefficient for inorganic nanoparticles and D_{qz} is the cross-diffusion coefficient. Furthermore, we proposed in [64] that the mass transport of monomer and short polymer is impeded by the inorganic nanoparticles, hence their respective diffusion fluxes can be modelled as

$$J_m = -D_m e^{-H_m z}\frac{\partial m}{\partial x}, \tag{2.9a}$$

$$J_p = -D_p e^{-H_p z}\frac{\partial p}{\partial x}. \tag{2.9b}$$

Although this model proved successful in predicting the redistribution of inorganic nanoparticles in holographic recording, in this work we prefer to explore the idea of cross-diffusion as a mutual or reciprocated process between the polymer and the inorganic nanoparticles. As the monomer molecules bind together to form short polymers, and as the short polymer bind together via a cross-linking monomer, any inorganic nanoparticle that were previously occupying the space between the monomer or short polymer are pushed out and accumulate where the concentration of polymer is minimal. The nanoparticle-polymer system is similar to a predator-prey model albeit the total quantity of nanoparticles is fixed. Earlier studies [65, 66, 67, 68] modelling cross-diffusion in predator-prey systems have expressed flux as the gradient

of a non-linear product. We propose that the flux for monomer, short polymer, cross-linked polymer and nanoparticles can be modelled as

$$J_m = -D_m \frac{\partial m}{\partial x},$$

$$J_p = -D_p \frac{\partial [p(1 + \epsilon_{pz} z)]}{\partial x},$$

$$J_q = -D_q \frac{\partial [q(1 + \epsilon_{qz} z)]}{\partial x},$$

$$J_z = -D_z \frac{\partial [z(1 + \epsilon_{pz} p + \epsilon_{qz} q)]}{\partial x}.$$

The constants ϵ_{ij} are called the cross-diffusion constants and are related to the interaction between species i and j. If i and j repel one another then $\epsilon_{ij} > 0$ while if i and j are attracted then $\epsilon_{ij} < 0$. We still hold the assumption that cross-linked polymer are immobile, $D_q = 0$. The new system of PDEs modelling the formation of holographic gratings in hybrid photopolymer media is

$$\frac{\partial m}{\partial t} = D_m \frac{\partial^2 m}{\partial x^2} - \Phi(t) F(x,t) m, \tag{2.10a}$$

$$\frac{\partial p}{\partial t} = D_p \frac{\partial^2 p}{\partial x^2} + \epsilon_{pz} D_p \frac{\partial^2 (pz)}{\partial x^2} + \Phi(t) F(x,t) m - \Phi(t) \Gamma p^2, \tag{2.10b}$$

$$\frac{\partial q}{\partial t} = \Phi(t) \Gamma p^2, \tag{2.10c}$$

$$\frac{\partial z}{\partial t} = D_z \frac{\partial^2 z}{\partial x^2} + \epsilon_{pz} D_z \frac{\partial^2 (pz)}{\partial x^2} + \epsilon_{qz} D_z \frac{\partial^2 (qz)}{\partial x^2}. \tag{2.10d}$$

If we define new nondimensionalized constants

$$\alpha_{mm} = \frac{D_m t_0}{\Lambda^2}, \qquad \alpha_{pp} = \frac{D_p t_0}{\Lambda^2}, \qquad \alpha_{pz} = \epsilon_{pz} z_0 \alpha_{pp},$$

$$\alpha_{zz} = \frac{D_z t_0}{\Lambda^2}, \qquad \alpha_{zp} = \epsilon_{pz} m_0 \alpha_{zz}, \qquad \alpha_{zq} = \epsilon_{qz} m_0 \alpha_{zz},$$

then the new nondimensionalized PDEs are

$$\frac{\partial m}{\partial t} = \alpha_{mm} \frac{\partial^2 m}{\partial x^2} - \Phi(t) \beta f(x,t) m, \tag{2.11a}$$

$$\frac{\partial p}{\partial t} = \alpha_{pp} \frac{\partial^2 p}{\partial x^2} + \alpha_{pz} \frac{\partial^2 (pz)}{\partial x^2} + \Phi(t) \beta f(x,t) m - \Phi(t) \gamma p^2, \tag{2.11b}$$

$$\frac{\partial q}{\partial t} = \Phi(t) \gamma p^2, \tag{2.11c}$$

$$\frac{\partial z}{\partial t} = \alpha_{zz} \frac{\partial^2 z}{\partial x^2} + \alpha_{zp} \frac{\partial^2 (pz)}{\partial x^2} + \alpha_{zq} \frac{\partial^2 (qz)}{\partial x^2}. \tag{2.11d}$$

The domain and boundary conditions remain unchanged from the original two-way diffusion model [7, 2]; for $0 \leq x \leq 1$ and $t \geq 0$

$$\frac{\partial m(0,t)}{\partial x} = \frac{\partial p(0,t)}{\partial x} = \frac{\partial q(0,t)}{\partial x} = \frac{\partial z(0,t)}{\partial x} = 0,$$
$$\frac{\partial m(1,t)}{\partial x} = \frac{\partial p(1,t)}{\partial x} = \frac{\partial q(1,t)}{\partial x} = \frac{\partial z(1,t)}{\partial x} = 0.$$

The initial conditions are

$$m(x,0) = 1, \qquad p(x,0) = 0, \qquad q(x,0) = 0, \qquad z(x,0) = z_0/m_0.$$

Numerical simulation of Eqn. 2.11 can be achieved with Crank-Nicolson implicit finite difference methods, see Appendix A.

2.5   Numerical Results

The predicted final distribution of cross-linked polymer, nanoparticles, and the time evolution of redistribution and RI modulation for a range of different positive values for cross-diffusion constants ϵ_{pz} and ϵ_{qz}, and for scattering constant ξ are presented in Figs. 2.3, 2.4, 2.5 and 2.6. The R code used to produce and plot these results can be found in Appendix C.

(a) Cross-linked polymer

(b) Nanoparticles

(c) Redistribution of nanoparticles

(d) RI modulation
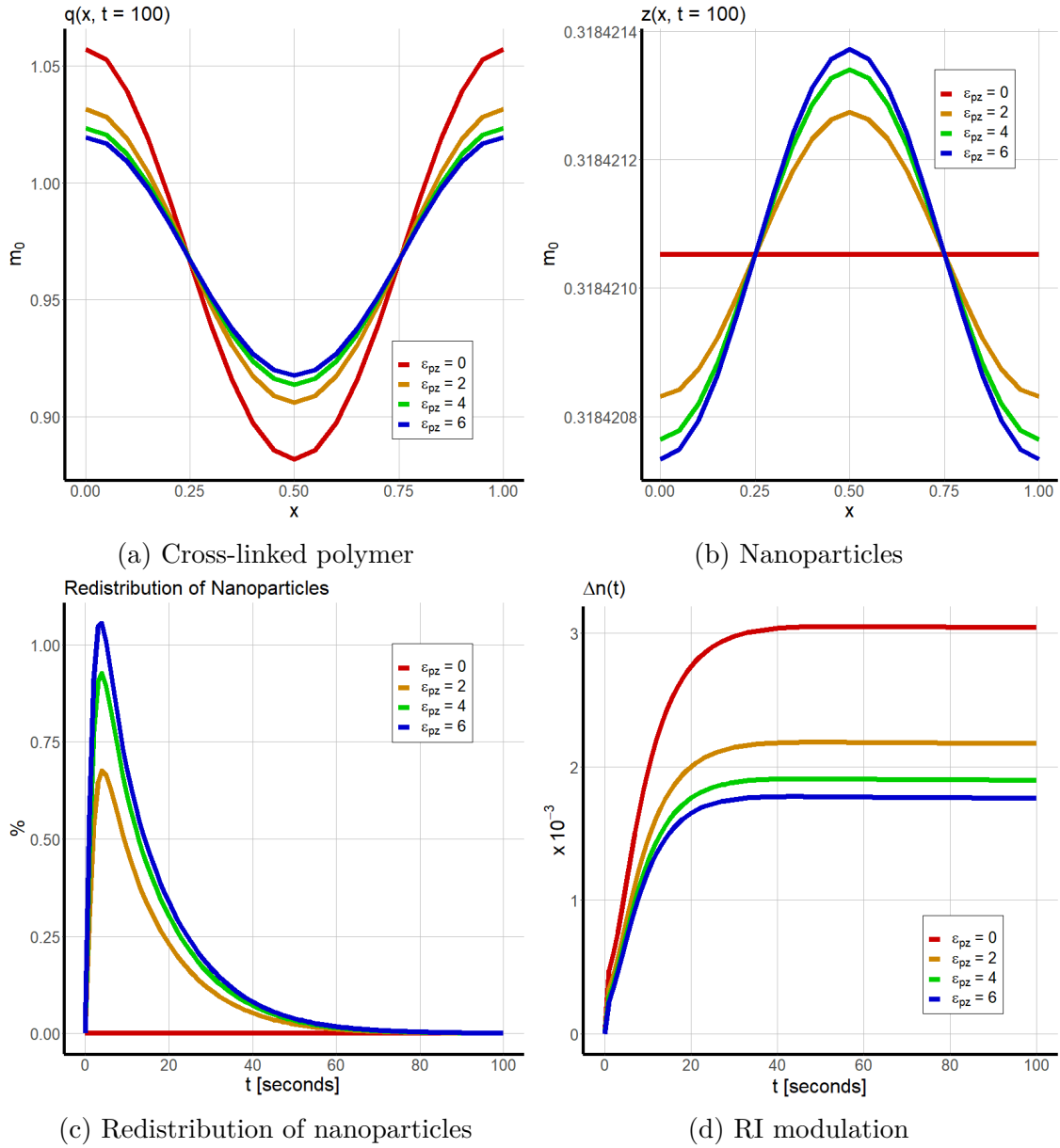
Fig. 2.3 The effect of ϵ_{pz} on the predicted grating profile after recording; $\xi = \epsilon_{qz} = 0$.

In simulations with positive ϵ_{pz} values the concentration gradient of short polymer will induce cross-diffusion of inorganic nanoparticles, redistribution of nanoparticles results in a concentration gradient which, subsequently, will induce cross-diffusion of short polymer. The high concentration of nanoparticles near the dark fringe push the mobile short polymer away trapping them near the bright fringes resulting in less non-local growth of cross-linked polymer and more near the point of initiation. This results in a greater concentration gradient of short polymer which in turn drives greater cross-diffusion of inorganic nanoparticles. However, it is only in the early stage of holographic recording that the concentration of short polymer exceeds that of the cross-linked polymer, eventually the latter will overtake the former. Towards the end of holographic recording all short polymer will be transformed into immobile cross-linked polymer, if $\epsilon_{qz} = 0$ then there is no cross-diffusion. Without any cross-diffusion keeping the nanoparticles at the dark fringe, self-diffusion takes over and

the grating profile of inorganic nanoparticles returns to a uniform geometry (see Fig. 2.3c). The predicted results for non-zero ϵ_{qz} are presented in Fig. 2.4.

(a) Cross-linked polymer

(b) Nanoparticles

(c) Redistribution of nanoparticles

(d) RI modulation
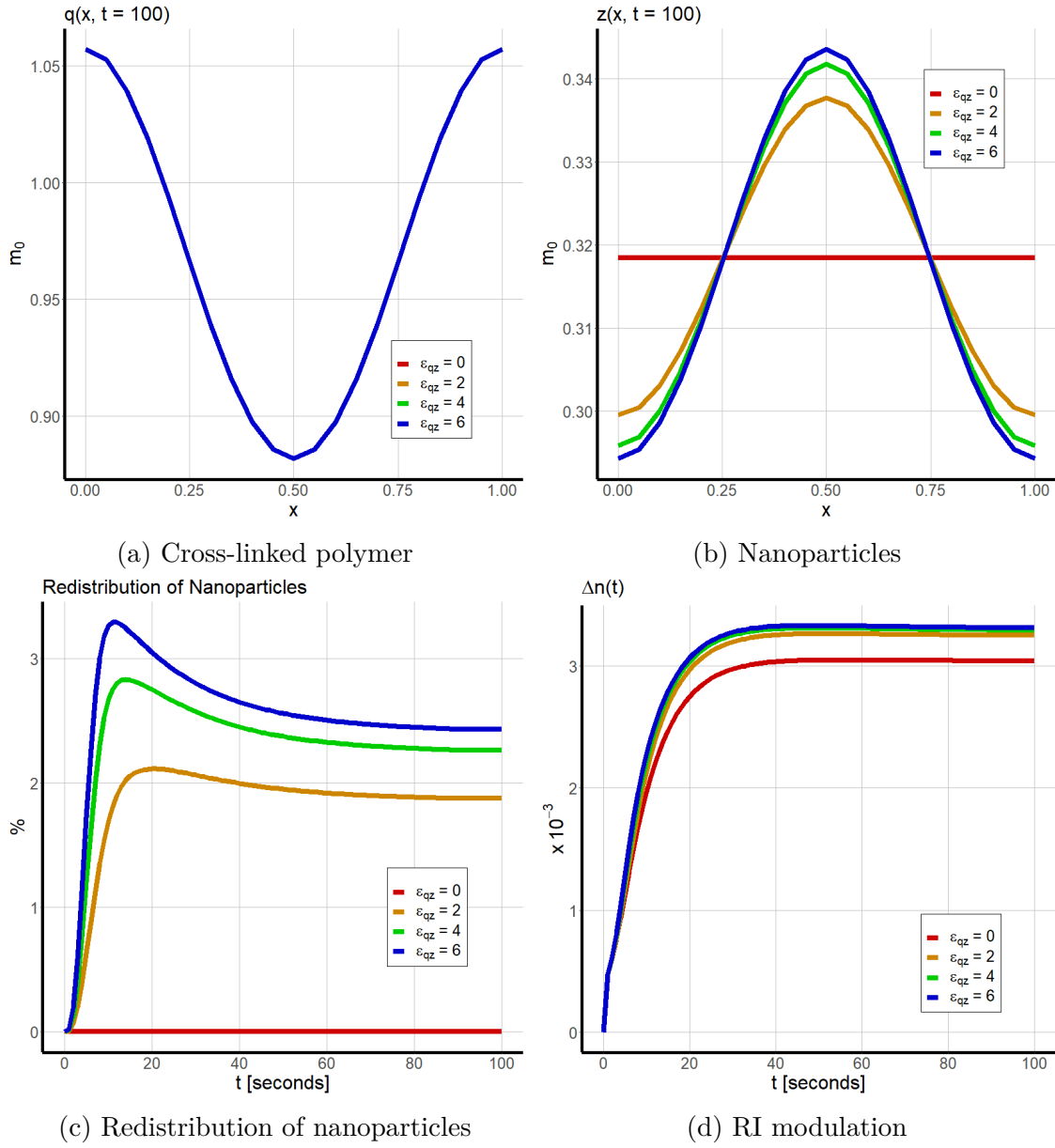
Fig. 2.4 The effect of ϵ_{qz} on the predicted grating profile after recording; $\xi = \epsilon_{pz} = 0$.

If $\epsilon_{pz} = 0$ the system is decoupled and the distribution of cross-linked polymer chains is not influenced by the redistribution of inorganic nanoparticles. With a non-zero ϵ_{qz} there is some redistribution of nanoparticles, as the concentration of cross-linked polymer near the point of initiation grows throughout holographic recording cross-diffusion pushes the nanoparticles towards the dark fringe. A concentration gradient of nanoparticles grows until self-diffusion overpowers cross-diffusion at which point redistribution begins to drop until an equilibrium is reached. If one assumes that polymer-nanoparticle interaction is the same for both short mobile polymers and immobile cross-linked polymers then $\epsilon_{pz} = \epsilon_{qz}$, the predicted results for this hypothesis are presented in Fig. 2.5.

(a) Cross-linked polymer

(b) Nanoparticles

(c) Redistribution of nanoparticles

(d) RI modulation
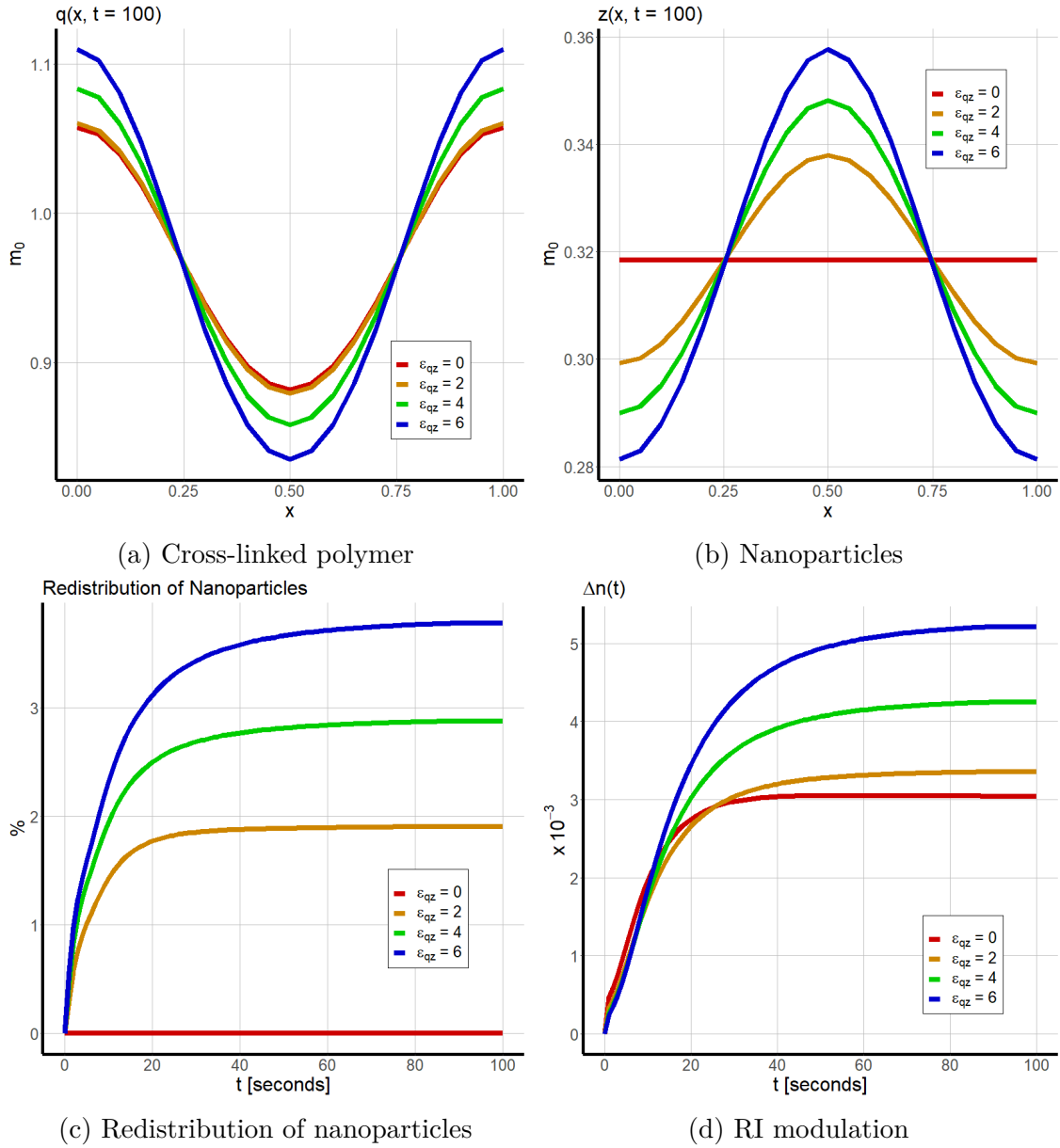
Fig. 2.5 The effect of ϵ_z on the predicted grating profile after recording; $\xi = 0$.

The combination of non-zero ϵ_{pz} and non-zero ϵ_{qz} means that there is both redistribution of nanoparticles and coupled diffusion between nanoparticles and short polymers. The growth of cross-linked polymer chains near the point of initiation creates a concentration gradient of $q(x, t)$ which induces cross-diffusion of nanoparticles. The assumption that the nanoparticle-polymer interaction does not distinguish between mobile short polymers and immobile cross-linked polymers has produced the best results in terms of predicted redistribution and inorganic nanoparticles. All subsequent numerical results in this thesis will use $\epsilon_{pz} = \epsilon_{qz} = \epsilon_z$. The accumulation of nanoparticles at the dark fringes creates a concentration gradient of $z(x, t)$ which induces cross-diffusion of short polymers. The short polymers are pushed away from the dark fringes which results in a greater concentration of cross-linked polymer at the bright fringes creating an even greater concentration gradient of $q(x, t)$. This leads to exponential growth in RI modulation as concentration gradients of nanoparticles and

cross-linked polymer drive one another. However, the redistribution of nanoparticles and polymer can be reigned in by the effects of photon scattering, see Fig. 2.6.

(a) Cross-linked polymer

(b) Nanoparticles

(c) Redistribution of nanoparticles

(d) RI modulation
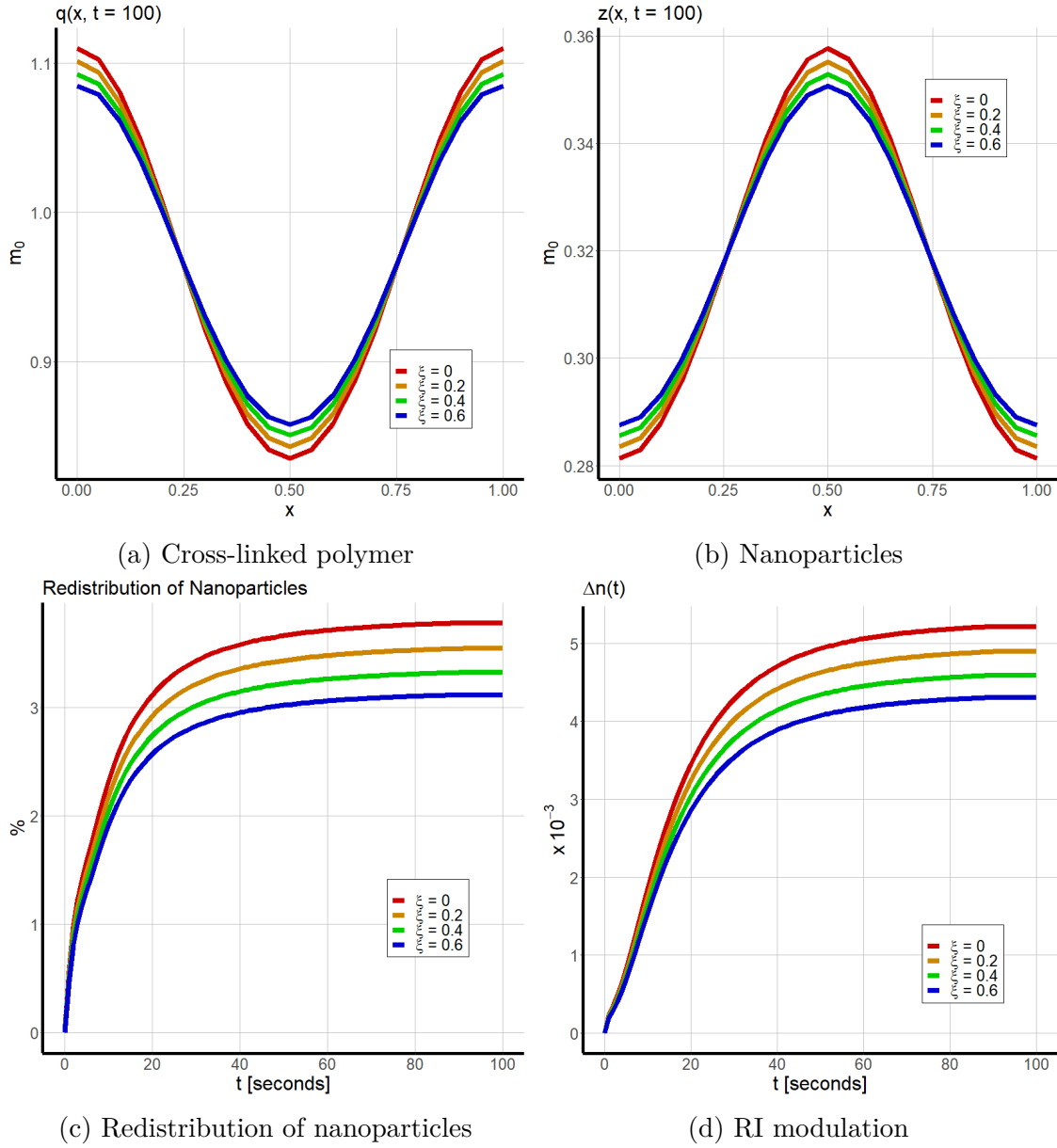
Fig. 2.6 The effect of ξ on the predicted grating profile after recording; $\epsilon_z = 6$.

Increasing ξ reduces the visibility of the interference pattern resulting in greater growth of short polymer, and hence cross-linked polymer, at dark fringe positions. A reduced concentration gradient of $q(x, t)$ means less cross-diffusion (and hence less redistribution) of inorganic nanoparticles and, consequently, a reduced RI modulation. Cross-diffusion traps the short polymer near the bright fringes while self-diffusion allows them to spread out, hence the effect of increasing dopant concentration should increase the concentration of cross-linked polymer chains near the bright fringe up to a point, beyond which increased dopant concentration results in greater non-local polymer growth. This suggests that there should be an optimal dopant concentration.

2.6   Reproduction of Experimental Results

The success of any mathematical model is measured by validating its predictions against experimental results. An earlier experimental study [55] in holographic recording made real-time measurements of RI modulation in hybrid photopolymer nanocomposites doped 0, 1, 2 and 5 wt.% with MFI nanozeolites. Their results showed a significant improvement in RI modulation in going from 0 to 1 wt.%, but diminishing returns when doping was increased to 2 wt.% and disimprovement when doping was further increased to 5 wt.%. The predicted RI modulation is plotted against nanoparticle doping in Fig. 2.7 for $\xi \in [0, 3.2]$ and $\epsilon_z \in [0, 15]$ alongside experimental results.

(a) Cross-diffusion fixed at $\epsilon_z = 13$ (b) Scattering fixed at $\xi = 1.6$
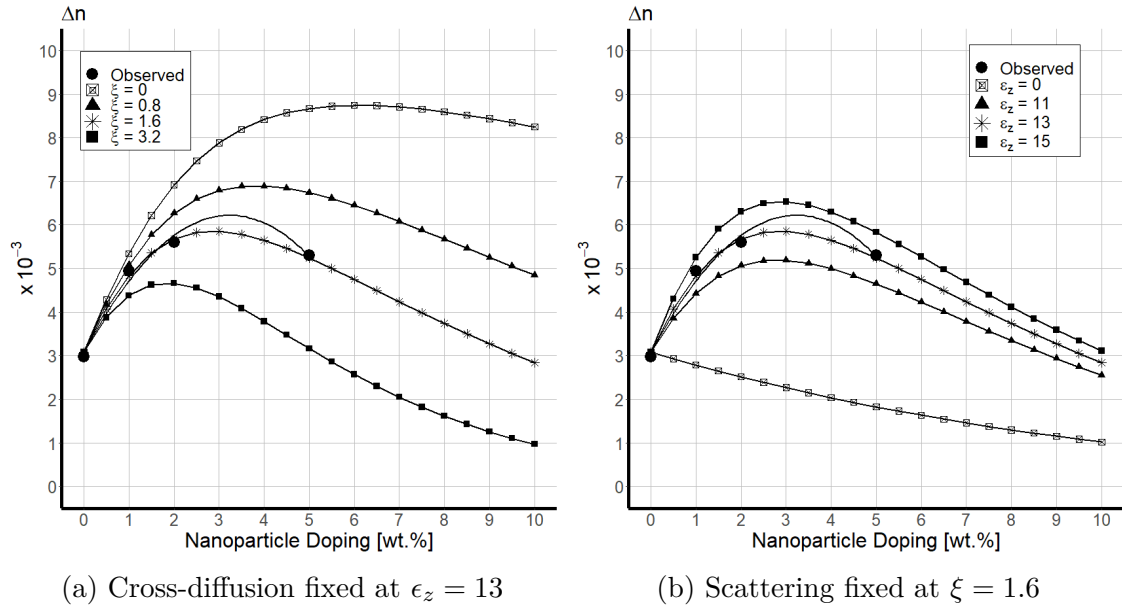
Fig. 2.7 The RI modulation at the end of holographic recording plotted against the initial nanoparticle dopant concentration for a range of different values for the scattering constant (left) and cross-diffusion constant (right).

Fig. 2.7a shows for a fixed cross-diffusion constant ($\epsilon_z = 13$) the pattern of significant improvement, diminishing returns and disimprovement in response to increasing doping of BEA nanozeolites from 0-5 wt.% is reproducible without scattering; however, optimal doping is around 6 wt.% which is much higher than the experimental results. By increasing scattering to $\xi = 1.6$ the predicted optimal doping occurs at 3 wt.%. In Fig. 2.7b the scattering constant is fixed ($\xi = 1.6$), without cross-diffusion the model cannot predict the large increase in RI modulation in going from 0 to 1 wt.%. A comparison of experimental results [55] for time evolution of RI modulation throughout a 100 second recording with model predictions is presented in Fig. 2.8.

(a) Observed [55]

(b) Predicted; $\xi = 1.6$, $\epsilon_z = 13$
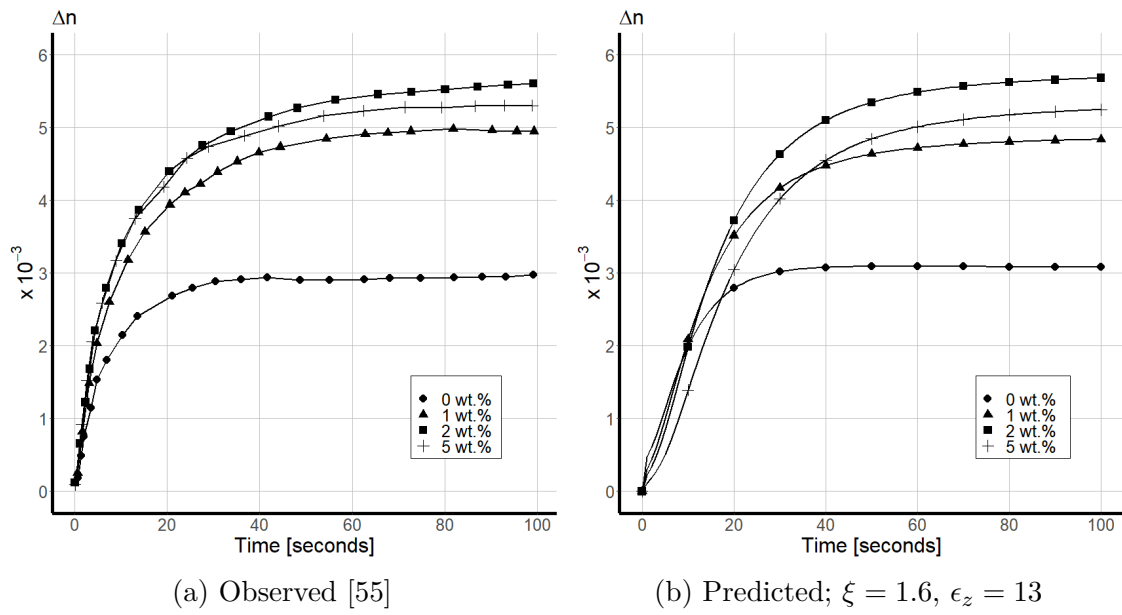
Fig. 2.8 A comparison of the observed and predicted change in Δn over 100 s exposure for dopant concentrations 0, 1, 2 and 5 wt.%.

The model predictions show reasonably good agreement with the experimentally observed behaviour. For a more in depth understanding of how the changes in doping effect the formation of the holographic grating the final grating profile of monomer, short polymer, cross-linked polymer and inorganic nanoparticles (BEA nanozeolites) is presented in Fig. 2.9.

(a) Monomer

(b) Short Polymer

(c) Immobilized Polymer
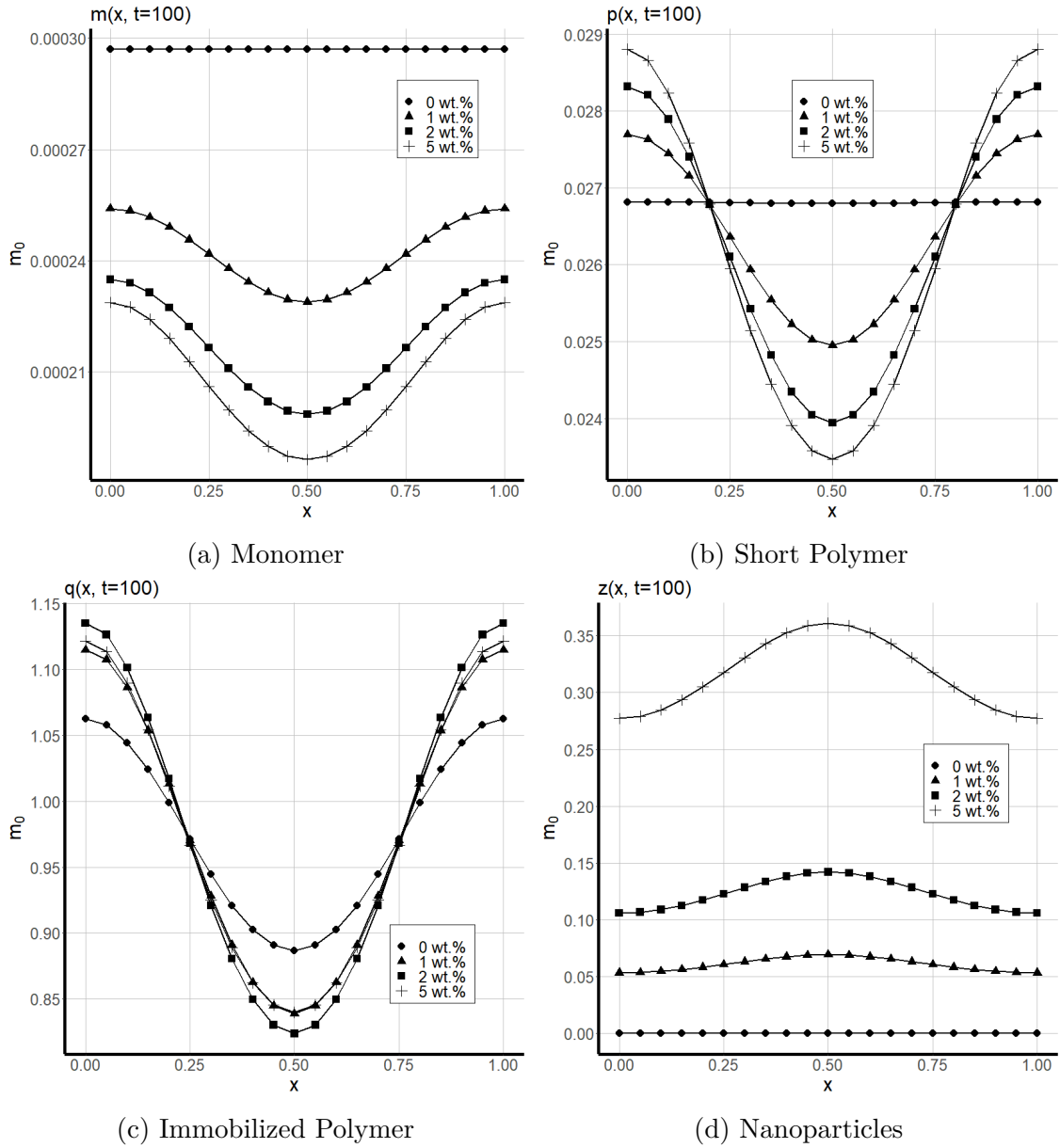
(d) Nanoparticles

Fig. 2.9 Final grating profile of monomer, short polymer, immobilized polymer and nanoparticles after a 100 s exposure for dopant concentrations 0 (circles), 1 (triangles), 2 (squares) and 5 (crosses) wt.%.

By the end of the 100 second recording almost all monomer and short polymer has been turned into immobilized polymer chains. Increasing doping from 0 to 1 wt.% results in significantly greater growth of cross-linked polymer at the bright fringes and significantly fewer polymer chains at the dark fringes as cross-diffusion induced by the concentration gradient of nanoparticles stymies the mass transport of short polymer. Increasing doping to 2 wt.% yields slightly more polymer at the bright fringe and slightly less at the dark fringe as scattering of light by nanoparticles reduces the contrast of the illumination pattern and begins to cancel out the effects of cross-diffusion. At 5 wt.% doping the effects of scattering are now dominant, the growth of cross-linked polymer at the bright fringes is even less than it was at 1 wt.% doping but due to the increased accumulation of low-RI nanoparticles at the dark fringe the drop in Δn in going from 2 to 5 wt.% is relatively small.

2.7 Grating Optimization

One of the aims of this research is to study how properties of the host photopolymer, the recording conditions and properties of the nanoparticles can be used to optimize the formation of a holographic grating for maximum RI modulation and copying accuracy. The conditions for holographic recording can be characterized by spatial frequency and rate of polymerization; in Figs. 2.10 and 2.11 the predicted final value of the Δn and distortion are plotted against the doping concentration. For holographic recording at spatial frequencies less than 400 lines/mm the fabricated grating falls, generally, within the planar phase geometry (determined by either the Klein-Cook or Moharam-Young criteria [69]). Planar and volume geometry gratings behave very differently to one another, the work in this chapter will restrict itself to the analysis of theoretically modelled volume phase holographic gratings. In all numerical simulations presented in this section the model constants are $\alpha_{pp} = 0.064$, $\gamma = 0.6$, $n_q = 1.64$, $b_0/m_0 = 5$, $\xi = 1.6$ and $\epsilon_z = 13$. These values proved effective in modelling the formation of holographic gratings in AA/PVA photopolymers doped with BEA nanozeolites, however, the coefficients associated with scattering and cross-diffusion are expected to be different for different nanoparticles depending on their size and optical properties.

(a) RI Modulation   (b) Distortion
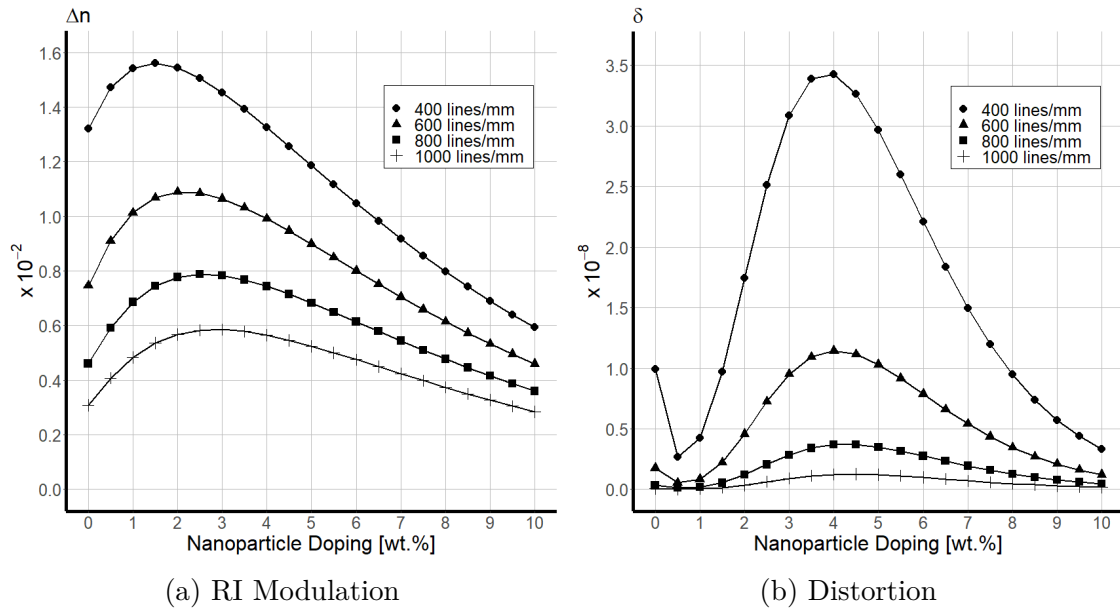
Fig. 2.10 RI modulation and distortion plotted against initial nanoparticle doping for spatial frequencies ranging from 400-1000 lines/mm at a fixed recording intensity of 0.5 mW/cm^2.

The simulation of low spatial frequency holographic recording in undoped photopolymers gives a greater predicted Δn. This is because at low spatial frequencies the distance between the bright fringes is larger so the mobile short polymer have a greater distance to cover. The addition of nanozeolites to the nanocomposite induces cross-diffusion of short polymer which results in a greater concentration of immobile cross-linked polymer at bright fringe positions. This drives greater cross-diffusion and

greater redistribution of nanozeolites which in turn induces further cross-diffusion of short polymer. RI modulation increases with increased doping until the effects of scattering become important. At the lowest spatial frequency simulated, 400 lines/mm, optimal performance was found at 1.5 wt.% while at the highest spatial frequency optimal performance was at 3 wt.%. In low spatial frequency recording monomer is consumed at bright fringe positions at a rate comparable with the rate at which it is restocked. Consequently, at any time t during recording there is a higher concentration of monomer at dark fringe positions $m(x = 0.5, t)$ compared with high spatial frequency recording conditions. This is why predicted distortion in low spatial frequency recording conditions is significantly greater compared with high spatial frequency conditions. There is a drop in distortion with the addition of a very small quantity (less than 1 wt.%) of nanozeolites. But further nanozeolite doping results in a steep increase in distortion due to the redistribution of nanozeolites. Beyond 4 wt.% the effects of scattering results in a drop in distortion with increased doping.

(a) RI Modulation   (b) Distortion
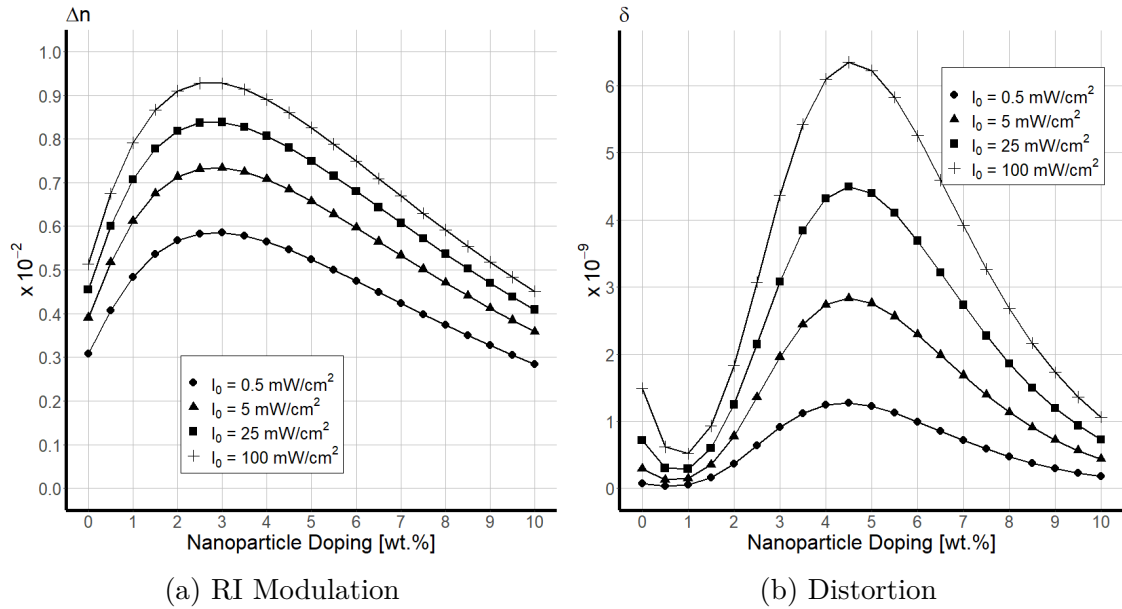
Fig. 2.11 RI modulation and distortion plotted against initial nanoparticle doping for recording intensities of 0.5-100 mW/cm^2 at a fixed spatial frequency of 1000 lines/mm.

Fig. 2.11a shows that higher recording intensities perform best albeit with substantial diminishing returns. Increasing intensity from 0.5 to 100 mW/cm^2 results in an improvement to the optimal performance at 3 wt.% of only 58%. Again, very low doping results in a drop in distortion; the redistribution of nanozeolites leads to increased distortion with increased doping up to 4 wt.% beyond which distortion drops again due to scattering.

A heatmap of the predicted Δn for a range of nanoparticle doping and nanoparticle RI is presented in Fig. 2.12; plots of RI modulation Δn and distortion δ against nanoparticle doping are presented in Fig. 2.13 for low and high RI nanoparticles. Photon scattering within the nanocomposite is a function of the difference between

the RI of nanoparticles and the RI of the surrounding material, $\xi = \xi(|n_b - n_z|)$, the results presented in Figs. 2.12 and 2.13 assume that scattering is constant with the range $|n_b - n_z| \leq 0.04$. This assumption would be valid for nanoparticles of size, typically less than 10 nm.
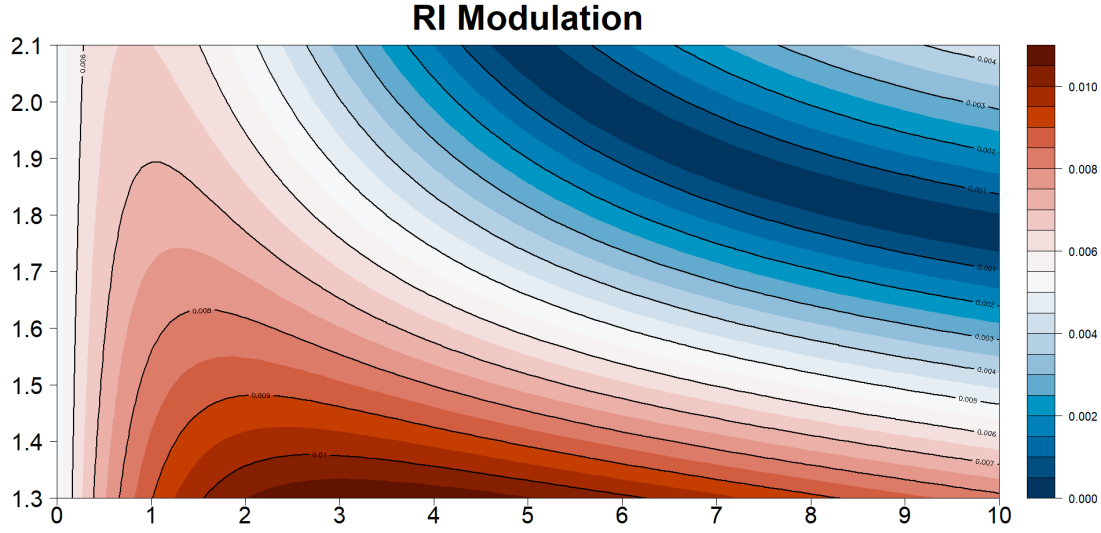
RI Modulation

Fig. 2.12 A heatmap of Δn plotted against nanoparticle wt.% on the x-axis and nanoparticle RI on the y-axis (1000 lines/mm).

(a) Δn
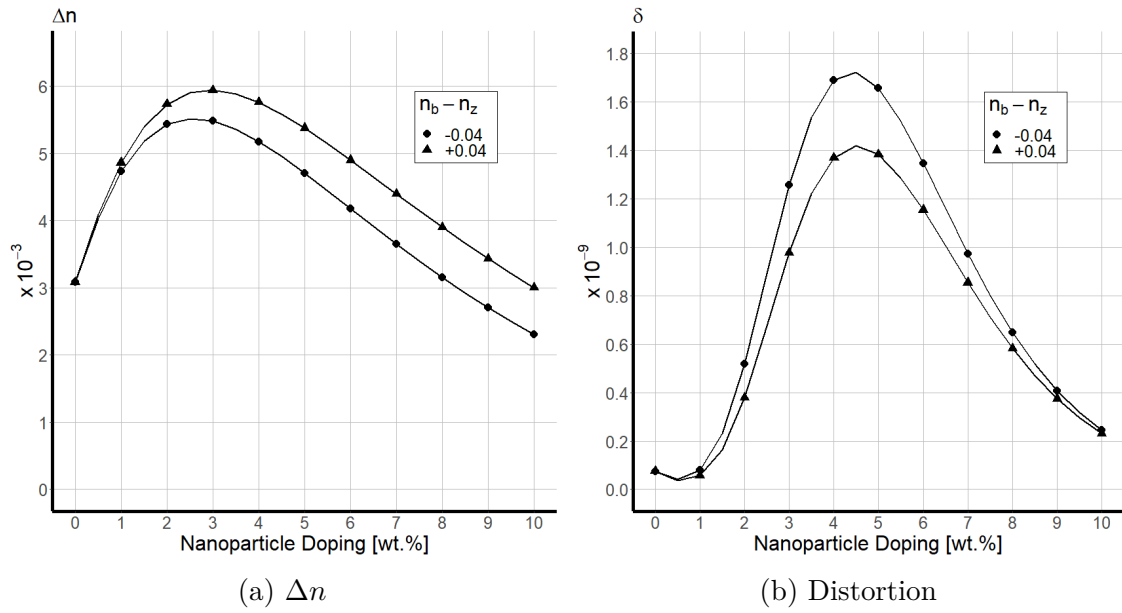
(b) Distortion

Fig. 2.13 RI modulation and distortion plotted against initial nanoparticle doping with low and high RI nanoparticles (1000 lines/mm).

One can see from both the heatmap and scatter plots that in weak polymerization conditions low-RI nanoparticles perform best according to model predictions both in terms of high RI modulation and low distortion with optimal performance achieved with nanoparticle doping of 2-4 wt.%.

2.8   Summary & Conclusions

Can a mathematical model predict the redistribution of inorganic nanoparticles in a hybrid photopolymer during holographic recording?

The model introduced in this chapter has been largely successful in achieving this research objective. A consequence of the addition of inorganic nanoparticles to the photopolymer matrix is photon scattering, the effect of which is reduction in the visibility of the recorded illumination pattern (Eqn. 2.1). The redistribution of inorganic nanoparticles in the recording of an unslanted holographic grating in a hybrid photopolymer material can be modelled by a system of coupled reaction-diffusion equations (Eqn. 2.10). Solutions to these PDEs can be used to model the fraction of inorganic nanoparticles that are redistributed in holographic recording (Eqn. 2.3). A comparison of experimental results with numerical simulation of the model presented in this chapter validates our hypothesis, namely that the mass transport of monomer, short polymer and inorganic nanoparticles are interdependent. Redistribution of nanoparticles is driven by a combination of self-diffusion and cross-diffusion induced by the concentration gradient of mobile short polymer and immobile cross-linked polymer. Experimental results [55] can be reproduced through numerical simulation of the model presented in this chapter with the appropriate selection of scattering and cross-diffusion constant. More specifically, the combination of mutual cross-diffusion and photon scattering results in an optimal doping beyond which there is a disimprovement in performance with increased doping. Numerical simulation of the mathematical model found optimal doping between 1-4 wt.% with the optimal value increasing with increased spatial frequency, also that low RI nanoparticle perform better in terms of high RI modulation and low distortion.

Chapter 3

Photopolymerization Induced Volume Shrinkage

One of the main caveats of photopolymers as a choice of media for holographic applications is the rotation of the Bragg angle and wavelength shift due to the change in volume of the photosensitive layer during recording. This effect is known as polymerization-induced shrinkage. Growing interest in photopolymer technology has motivated a great deal of investigation into understanding the phenomena and developing techniques to minimize the shrinkage in holographic recording, however, it is not possible to negate shrinkage in all recording geometries. Shrinkage occurs because the chemical reaction that results in the conversion of monomer molecules to polymer chains requires van der Waals bonds to be replaced with covalent bonds the consequence of which is a loss in free volume [70]. For some holographic applications (holographic displays, optical elements, data storage, etc.) shrinkage is undesirable because there is significant distortion in the reconstructed image and additional counter agents must be added to correct the output, the upper limit of shrinkage for commercial viability is 0.5% [71]. For holographic sensing applications shrinkage can be advantageous, by measuring the change in volume, the concentration of a target analyte can be quantified. The cross-linking of short polymer squeezes air and water molecules out leaving a more compact solid polymer layer. The transformation of the initial solution of monomer to the more closely packed polymer is represented in the mathematical model as a higher density, $\rho_p > \rho_m$. A consequence of the high density is that the volume occupied by immobile polymer chains is less than the volume occupied by the monomer. The thickness of the samples on which holographic gratings are recorded is infinitesimal in comparison with the length and width, consequently, any significant change in volume takes place in the thickness of the nanocomposite.

This chapter aims to address the following research question: *can the formation of slanted holographic gratings be modelled mathematically such that published experimental results, specifically the reduced shrinkage with increased nanoparticle doping [59] and increased shrinkage at high spatial frequencies [72], can be predicted*

theoretically? To that end we will begin with an overview of modelling shrinkage in unslanted holographic gratings with our analysis restricted to a single spatial dimension. Through numerical simulation of the existing equations, we can show why it is necessary to extend the analysis to a second spatial dimension and how the model described in the previous chapter can be augmented further for this purpose. Extending the analysis to two spatial dimensions requires new transformation of variables and new boundary conditions. Furthermore, a new calculation is required for the predicted optical properties of theoretically modelled slanted holographic gratings. We will end the chapter by outlining how our model can predict the geometry of a theoretically modelled holographic grating.

3.1 Shrinkage in Unslanted Holographic Gratings

It is possible to calculate shrinkage in theoretically modelled unslanted holographic gratings if we know the mass densities of the components of the nanocomposite and the concentration of each part as a function of its position in the grating x and time t. The volume of a holographic grating averaged over the spatial domain can be expressed as

$$v(t) = \frac{1}{\Lambda} \int_0^\Lambda \left[\frac{m(x,t)}{\rho_m} + \frac{p(x,t)}{\rho_p} + \frac{q(x,t)}{\rho_q} + \frac{z(x,t)}{\rho_z} + \frac{b_0}{\rho_b} \right] \, dx,$$

and the initial volume of the nanocomposite is

$$v(0) = \frac{1}{\Lambda} \int_0^\Lambda \left[\frac{m_0}{\rho_m} + \frac{z_0}{\rho_z} + \frac{b_0}{\rho_b} \right] \, dx.$$

Shrinkage in the nanocomposite is equal to the fractional change in volume

$$\text{Volume Shrinkage} = \frac{v(t) - v(0)}{v(0)}.$$

Earlier studies [72] of holographic recording in acrylamide based photopolymer media measured volume shrinkage in the photosensitive layer via Bragg detuning of transmission diffraction gratings recorded at different slant angles, beam intensities, layer thicknesses and spatial frequencies. The observed shrinkage [72] at spatial frequencies 500, 1000, 1500 and 2000 lines/mm and recording intensities 1, 5 and 10 mW/cm^2 is presented in Fig. 3.1a. The exposure duration of the recording was adjusted so as to maintain a constant energy input of 80 mJ/cm^2. The conclusions of this study are clear, polymerization induced volume shrinkage in the photosensitive layer is minimized at high recording intensities and low spatial frequencies. By comparison, results for polymerization induced shrinkage predicted with our mathematical model are presented in Fig. 3.1b.

(a) Observed [72]   (b) Predicted
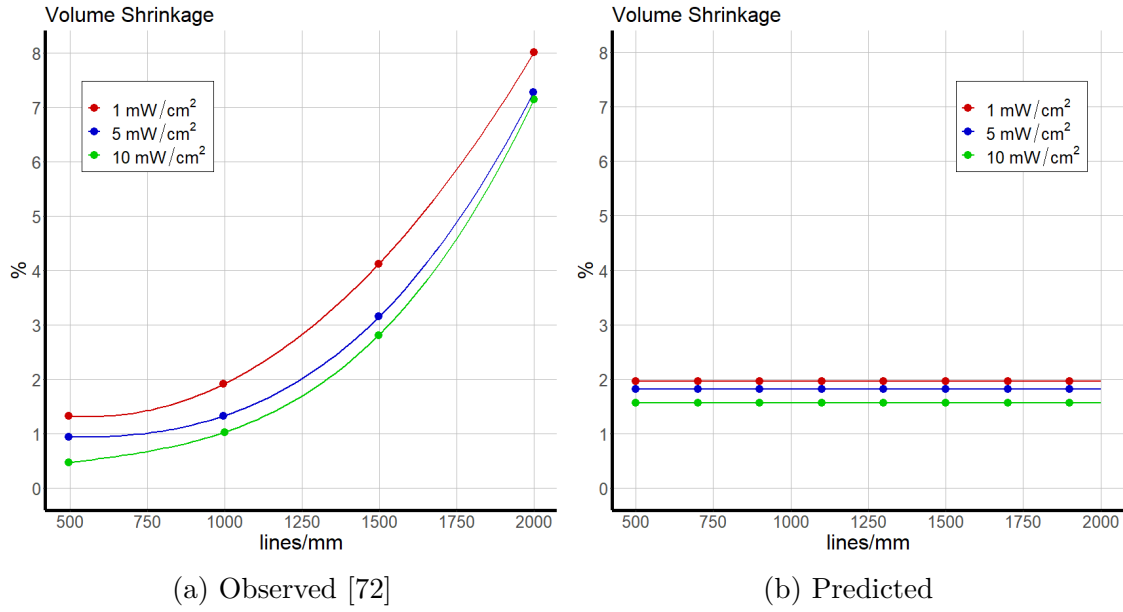
Fig. 3.1 The influence of spatial frequency and recording intensity on polymerization induced volume shrinkage in an undoped acrylamide photopolymer.

There is mixed success, the model can predict the increased shrinkage at low recording intensities, this is because when there are fewer photons incident on the sample per unit time, the rate at which monomer is transformed into mobile short polymer is reduced. Diffusion of mobile short polymer formed at the bright fringes moves them towards the dark fringes where the probability of cross-linking is minimal. At higher immobilization rates cross-linking of short polymer at the bright fringes takes place at a faster rate than its diffusion, consequently, a greater fraction of the initial sample is transformed into immobile cross-linked polymer chains. If $\rho_q > \rho_p$ this will result in increased volume shrinkage. However, the model fails to predict the increased shrinkage at high spatial frequencies. There are three factors that will determine the predicted shrinkage in photopolymerization

1. The fraction of the initial volume taken up by the monomer; holographic recordings by IEO typically use 17:83 monomer-binder volume ratio.

2. The fraction of monomer converted to polymer; published results relating to shrinkage in holographic recording [72] tend to use data collected after all monomer in the sample has been polymerized.

3. The increase in density resulting from the change in morphology due to polymerization.

The predicted shrinkage from numerical simulations of the model introduced in Chapter 2 run using an initial binder-monomer ratio b_0/m_0 ranging from 1 to 6 and the change in morphology $\rho_q - \rho_m$ ranging from 0 to 1.2 g/cm^3 are shown in Fig. 3.2.

(a) Binder-Monomer Ratio

(b) Difference in density of polymer and monomer
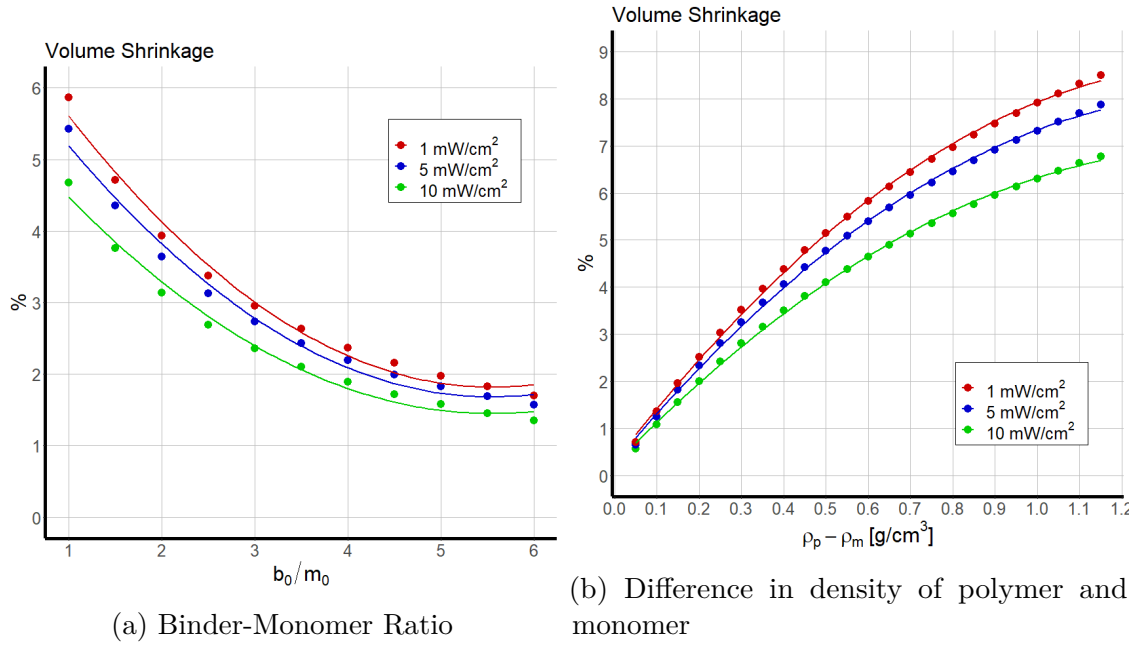
Fig. 3.2 Polymerization induced volume shrinkage in an AA photopolymer; 0 wt.%; 80 mJ/cm^2.

Fig. 3.2a shows that for fixed mass densities $\rho_m = 1.15$ g/cm^3 and $\rho_q = 1.3$ g/cm^3, predicted shrinkage is greater when monomer accounts for a greater portion of the initial volume. The PVA binder accounts for 83% of the initial volume for the holographic recordings in Fig. 3.1a.

Numerical simulations suggest that the 8% shrinkage observed at 2000 lines/mm would require the binder to account for a very small fraction of the initial volume if the change in density is as reported in [7]($\rho_q - \rho_m = 0.15$ g/cm^3). These simulations assume that the monomer is the only component of the nanocomposite to undergo a change in volume during polymerization; if the assumption that the binder can also change volume by polymerization is acceptable then the observed 8% shrinkage would be possible with only a small increase in polymer density. The results in Fig. 3.2b are from numerical simulations with binder taking up 83% of the initial volume and assuming that monomer density is fixed at $\rho_m = 1.15$ g/cm^3, the model predicts that very high shrinkage is possible if the density of immobilized polymer is $\rho_q = 2.35$ g/cm^3. The increased shrinkage at high spatial frequencies may be due to an decrease in rate of termination resulting in the production of longer polymer chains and thus greater loss of van der Waals volume as weak inter-molecular interactions as replaced with covalent bonds.

The idea that the dynamic range of diffraction efficiency in holographic gratings could be improved through the addition of inorganic nanoparticles with a RI either substantially higher or lower than that of the polymer was pioneered by [63, 44, 73, 74]. It has also been demonstrated [43] that doping photopolymers with inorganic nanoparticles can reduce the change in volume during photopolymerization. This phenomenon is expected because inorganic nanoparticles are inert and have a high

elastic modulus, they will not change volume during photopolymerization. With nanoparticles added to the photopolymer mixture the monomer takes up a smaller fraction of the initial volume. As illustrated in Fig. 3.2a the predicted change in volume is directly proportional to the monomer initial volume fraction. Earlier studies at IEO [75, 59] reported the observed change in volume in hybrid photopolymers with different levels of doping with MFI nanozeolites; Fig. 3.3 shows how the predicted shrinkage compares with the empirical results.
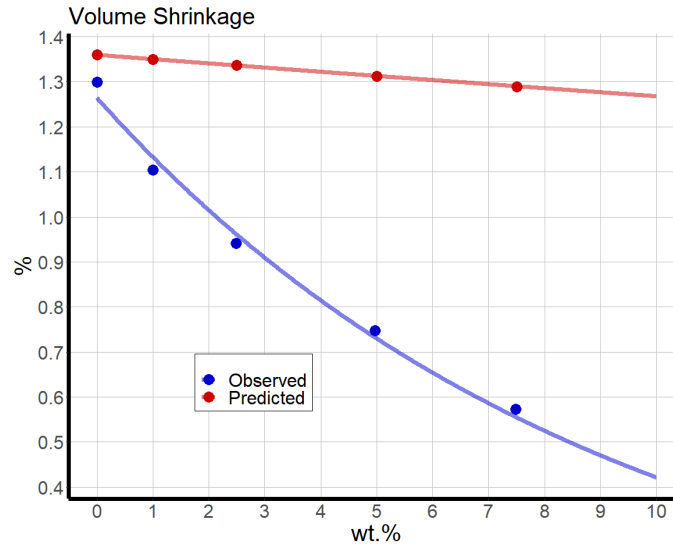
Fig. 3.3 Change in shrinkage with increased doping; $\rho_q = 1.25$ g/cm^3

Although the model has successfully predicted a drop in shrinkage with increased doping a comparison of the y-axis scale shows the predicted drop in shrinkage is much too small. It's possible that the presence of inorganic nanoparticles may disrupt the propagation stage of photopolymerization. Radicalized monomer need to be in proximity to one another to bind together and form a growing polymer chain, large nanoparticles increase the mean distance between monomer molecules resulting in early termination and the formation of shorter polymer chains with a smaller molecular mass. Another challenge associated with photopolymers as a choice of media for holographic applications are the real-object effects that arise from the finite thickness of the sample. The mathematical model assumes that diffusion of monomer and mobile short polymer is restricted to the x-axis, the direction normal to the Bragg plane and that the Bragg planes are normal to the surface of the sample, but this assumption is not a realistic one. In reality, the photosensitive layer is a real three dimensional object with finite depth. As a photon passes through a photopolymer system, the greater the thickness of the photopolymer system, the more opportunities there are for the photon to interact with a dye molecule and initiate photopolymerization. In the reconstruction stage, the greater the thickness of the grating, the more planes of periodically varying RI a visible wavefront must pass through and consequently transfer more optical power to the diffracted beam. Mathematical models to date describing the formation of holographic gratings have

restricted their analysis to the single dimensional length across the grating period, however, modelling photopolymerization induced shrinkage will require the equations from Chapter 2 to be extended to a second spatial dimension. This will require changes to both the reaction and diffusion terms and how the RI modulation is modelled. There are two additional effects unique to a three dimensional grating which should be taken into account in a realistic model; the absorption of light as it penetrates the depth of the sample and how the intensity peak of the illumination pattern shifts within the depth of slanted gratings.

3.2   Modelling Slanted Gratings

In experimental studies of shrinkage in holographic recording, the fractional change in material thickness is calculated via the fringe-plane rotation model. Consider two coherent visible wavefronts with wavelength λ_r which interfere in a medium with RI n_0 at angles of incidence θ_1 and $\theta_2 \neq \theta_1$, the resultant illumination pattern will have a grating period [3]

$$\Lambda = \frac{\lambda_r}{2n_0 \sin{(\phi + \theta_B)}},$$

whereby

$$\phi = \frac{\theta_1 - \theta_2}{2} \qquad \text{and} \qquad \theta_B = \frac{\theta_1 + \theta_2}{2}.$$

The former, ϕ, is the slant angle (see Fig. 1.1). When a volume phase (thick) transmission hologram is illuminated with light from a monochromatic source, it will only diffract at a specific angle known as the Bragg angle, θ_B. Reflection holograms behave differently, they can be illuminated with white light and diffract only one specific color (wavelength). An optical wavefront travelling through the holographic material at an angle of incidence ϕ is refracted towards the plane of normal at the point of incidence, the refracted slant angle is

$$\phi_r = \sin^{-1}\left[\frac{\sin \phi}{n_0}\right].$$

If the recording medium is attached to a rigid substrate (usually glass), and the elastic modulus of polymer chains to sufficiently large that there is no physical bending of the Bragg planes, then there is a change in volume of the recording medium which requires the planes with high concentrations of immobilized polymer chains to rotate and become more closely spaced resulting in a new grating period and a new slant angle (Fig. 3.4).
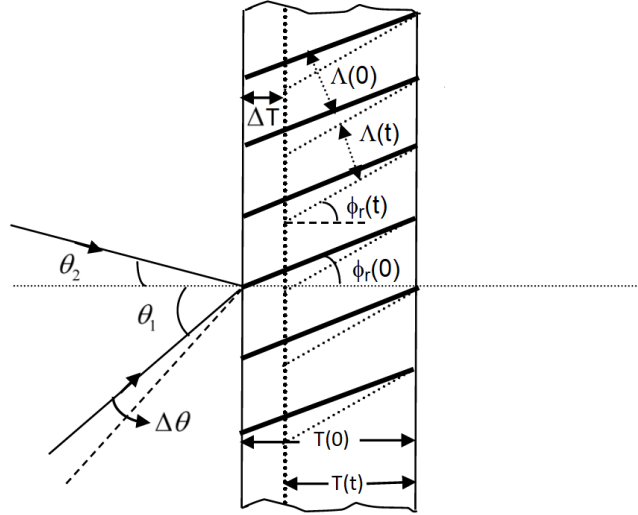
Fig. 3.4 Bragg planes at the start of holographic recording and at time t [4].

As the volume changes throughout holographic recording, both the grating period, thickness and slant angle must be expressed as a function of time; $\Lambda = \Lambda(t)$, $T = T(t)$ and $\phi_r = \phi_r(t)$. If the substrate to which the holographic material is fixed is a rigid body then one can assume that the point of contact between the holographic fringe and the glass backing does not change due to shrinkage, therefore

$$T(t) \tan \phi_r(t) = T_0 \tan \phi_r(0) \qquad \text{and} \qquad \frac{\Lambda(t)}{\cos \phi_r(t)} = \frac{\Lambda(0)}{\cos \phi_r(0)}. \qquad (3.1)$$

In experimental studies, shrinkage is measured via the detuning of the Bragg angle, consequently, it would not be possible to measure shrinkage in a perfectly unslanted holographic grating. Thus far, the mathematical framework has been restricted to modelling the formation of *unslanted* holographic gratings, i.e. the plane of the bright fringes of the interference pattern, also known as the Bragg plane, is normal to the surface of the recording material. This section will discuss how the mathematical model can be further augmented to predict and explain the behaviour of slanted gratings recorded in hybrid photopolymer media. As before, this will require numerical solutions to a coupled system of partial differential equations to yield spatial-temporal grating profiles of the concentration of the monomer, etc., but where previously the model only needed the relative concentration along one direction, modelling slanted gratings will need to look at the concentrations along two directions, the length and depth. The x and y-directions are defined as parallel and normal, respectively, to the surface of the recording material as illustrated in Fig. 3.5.
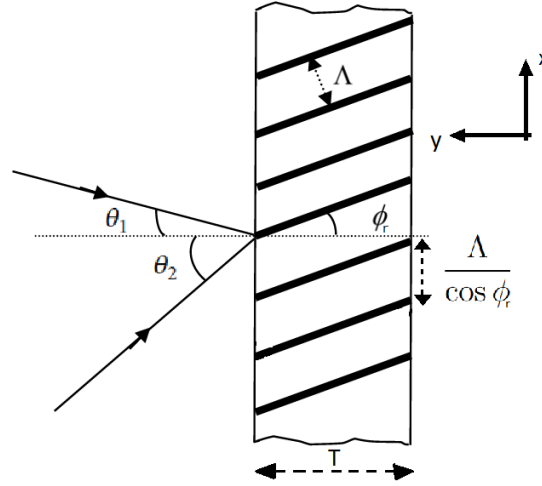
Fig. 3.5 A slanted holographic grating [4].

The y-axis is defined such that $y = 0$ at the fixed surface (the right in Fig. 3.5) and $y = T$ at the free surface. The refracted slant angle, ϕ_r, is important for defining the nondimensionalization of the two spatial dimensions. Mass transport in the y-direction requires a concentration gradient across the depth of the photosensitive layer, this can happen if there is spatial non-uniformity in the polymerization function, $F = F(x, y, t)$, due to absorption of light. In any light-matter interaction there will be a transfer of electromagnetic energy carried by a photon into thermal energy in the atoms and molecules, this phenomenon is called absorption and occurs when the quanta of energy carried by the photon matches the energy difference in the quantum mechanical states of the constituent molecules. The Beer-Lambert law [76] describes the drop in light intensity across the depth of the holographic material,

$$I(y) = I_0 e^{-\zeta(T-y)}.$$

The probability of absorption is measured by an absorption coefficient ζ which is a function of the chemical composition, physical state of the matter and the wavelength spectrum of the light. If there is significant absorption of light in a photopolymer system the photons cannot reach the deeper layers of the sample, consequently, all photopolymerization and immobilization takes place near the surface, the deeper layers of the sample serve as a reservoir of monomer. Earlier studies in holographic recording in photopolymer media have found that the recording beam intensity has a significant impact on the rate of polymerization [40, 39]. Changes are needed for Eqn. 2.1 to incorporate the absorption of photons in the photosensitive layer and how the shape of the illumination profile changes at different depths within the structure.

$$F(x, y, t) = k_p \left[I_0 e^{-\zeta(T-y)} \right]^a \left\{ 1 + V e^{-\xi z} \cos \left[\frac{2\pi \cos \phi_r(t)}{\Lambda(t)} x - \frac{2\pi \sin \phi_r(t)}{\Lambda(t)} y \right] \right\}.$$

The new expression in the square brackets allow the pattern to repeat itself in the x-direction over a distance $\Lambda/\cos\phi_r$ and in the y-direction over a distance $\Lambda/\sin\phi_r$. The visibility is a function of the intensity of the two writing beams (Eqn. 1.1); incorporating the absorption function into the calculation shows that visibility is independent of depth,

$$
V(y) = \frac{2\sqrt{I_1 I_2}}{I_1 + I_2} = \frac{2\left[I_1^{(0)} I_2^{(0)} \exp\left(-2\zeta y \cos\phi\right)\right]^{1/2}}{\left[I_1^{(0)} + I_2^{(0)}\right]\exp(-\zeta y \cos\phi)} = \frac{2\sqrt{I_1^{(0)} I_2^{(0)}}}{I_1^{(0)} + I_2^{(0)}} = V; \qquad V \in \mathbf{R}.
$$

New characteristic lengths defining the periodicity of the illumination pattern are required. Define $\hat{x}$ and $\hat{y}$ as the distance along the x- and y-directions respectively at which the illumination pattern projected onto the x-direction overlaps perfectly with the pattern at the fixed surface. These parameters are calculated as

$$
\hat{x} = \frac{\Lambda(t)}{\cos\phi_r(t)} = \frac{\Lambda(0)}{\cos\phi_r(0)}, \qquad\qquad \hat{y}(t) = \frac{\Lambda(t)}{\sin\phi_r(t)}. \qquad (3.2)
$$

The glass backing of the recording material does not permit movement of polymer chains in the x direction, hence, $\hat{x}$ will remain a constant throughout holographic recording. The x-axis is defined such that $x = 0$ at a point where the Bragg plane meets the fixed surface. Hence, we have a periodic spatial domain with period $\hat{x}$. A visual definition of the new characteristic lengths is illustrated in Fig. 3.6; the colored lines in Fig. 3.6a are a top-down view of the curves plotted in Fig. 3.6b, the dashed black line represents the Bragg plane and is the intersection of the peaks of the illumination peaks along the x-direction. The corresponding y-direction profile is shown in Fig. 3.7.

(a) A "top-down" view of a slanted grating.

(b) $F(x, y, t)$ vs x
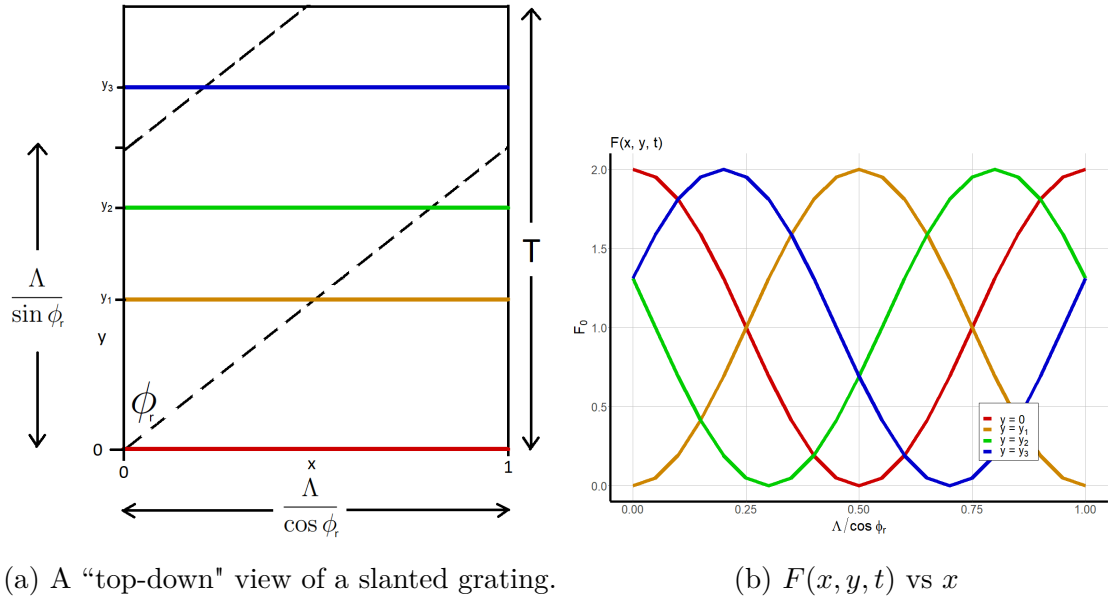
Fig. 3.6 A slanted holographic grating; the dashed line represents the Bragg plane; both graphics use a red curve for $y = 0$, orange for $y = y_1$, green for $y = y_2$ and blue for $y = y_3$ ($\zeta = 0$ cm^{-1}).

(a) A "top-down" view of a slanted grating.

(b) $F(x, y, t)$ vs y
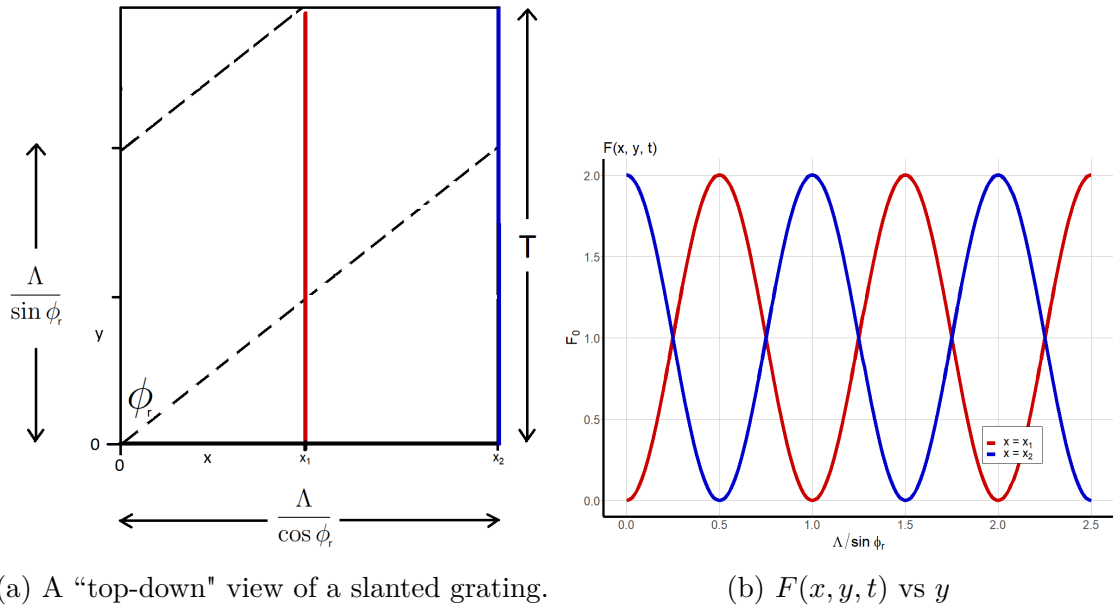
Fig. 3.7 A slanted holographic grating; the dashed line represents the Bragg plane; both graphics use a red curve for $x = x_1$ and blue for $x = x_2$ ($\zeta = 0$ cm^{-1}).

There is mass transport of monomer, short polymer and inorganic nanoparticles in two spatial dimensions, hence the respective diffusion fluxes are now expressed as vectors,

$$\vec{J_m} = -D_m \frac{\partial m}{\partial x}\vec{i} - D_m \frac{\partial m}{\partial y}\vec{j},$$

$$\vec{J_p} = -D_p \left\{ \left[\frac{\partial p}{\partial x}\vec{i} + \frac{\partial p}{\partial y}\vec{j} \right] + \epsilon_{pz} \left[\frac{\partial (pz)}{\partial x}\vec{i} + \frac{\partial (pz)}{\partial y}\vec{j} \right] \right\},$$

$$\vec{J_z} = -D_z \left\{ \left[\frac{\partial z}{\partial x}\vec{i} + \frac{\partial z}{\partial y}\vec{j} \right] + \epsilon_{pz} \left[\frac{\partial (pz)}{\partial x}\vec{i} + \frac{\partial (pz)}{\partial y}\vec{j} \right] + \epsilon_{qz} \left[\frac{\partial (qz)}{\partial x}\vec{i} + \frac{\partial (qz)}{\partial y}\vec{j} \right] \right\}.$$

The mass concentration of monomer, short polymer, cross-linked polymer and inorganic nanoparticles are solutions to

$$\frac{\partial m}{\partial t} + \nabla \cdot \vec{J_m} = -\Phi(t)F(x,y,t)m, \tag{3.3a}$$

$$\frac{\partial p}{\partial t} + \nabla \cdot \vec{J_p} = \Phi(t)F(x,y,t)m - \Phi(t)\Gamma p^2, \tag{3.3b}$$

$$\frac{\partial q}{\partial t} = \Phi(t)\Gamma p^2, \tag{3.3c}$$

$$\frac{\partial z}{\partial t} + \nabla \cdot \vec{J_z} = 0, \tag{3.3d}$$

on the domain $0 \le x \le \hat{x}$, $0 \le y \le T(t)$ and $t \ge 0$. As there is no net transfer of matter into, nor out of, the spatial domain the y domain is subject to a zero-flux boundary condition while the x domain is subject to a periodic boundary condition. The initial conditions are

$$m(x,y,0) = m_0, \qquad p(x,y,0) = 0, \qquad q(x,y,0) = 0, \qquad z(x,y,0) = z_0. \tag{3.4}$$

The variables m, p, q and z are all periodic on the x domain, i.e. $m(x+\hat{x},y,t) = m(x,y,t)$, etc. A periodic boundary can be modelled by the condition [77]

$$\frac{\partial^n m}{\partial x^n}(0,y,t) = \frac{\partial^n m}{\partial x^n}(\hat{x},y,t) \qquad\qquad n = \{0,1,2,...\}, \tag{3.5a}$$

$$\frac{\partial^n p}{\partial x^n}(0,y,t) = \frac{\partial^n p}{\partial x^n}(\hat{x},y,t) \qquad\qquad n = \{0,1,2,...\}, \tag{3.5b}$$

$$\frac{\partial^n z}{\partial x^n}(0,y,t) = \frac{\partial^n z}{\partial x^n}(\hat{x},y,t) \qquad\qquad n = \{0,1,2,...\}. \tag{3.5c}$$

There is no movement of material across the boundary at $y = 0$ nor at $y = T$. The zero-flux boundary condition can be applied to the y domain

$$\frac{\partial m}{\partial y}(x,0,t) = \frac{\partial p}{\partial y}(x,0,t) = \frac{\partial q}{\partial y}(x,0,t) = \frac{\partial z}{\partial y}(x,0,t) = 0, \tag{3.6a}$$

$$\frac{\partial m}{\partial y}(x,T,t) = \frac{\partial p}{\partial y}(x,T,t) = \frac{\partial q}{\partial y}(x,T,t) = \frac{\partial z}{\partial y}(x,T,t) = 0. \tag{3.6b}$$

The thickness of the nanocomposite can be expressed as a function of time by first establishing an expression for the volume. The initial mass concentrations of monomer, binder and inorganic nanoparticles are defined such that at $t = 0$ on the domain $[0, \hat{x}] \times [0, T_0]$

$$m_0 = \frac{1}{\hat{x}T_0} \int_0^{T_0} \int_0^{\hat{x}} m(x, y, 0) \; dx \; dy, \tag{3.7}$$

$$b_0 = \frac{1}{\hat{x}T_0} \int_0^{T_0} \int_0^{\hat{x}} b(x, y, 0) \; dx \; dy, \tag{3.8}$$

$$z_0 = \frac{1}{\hat{x}T_0} \int_0^{T_0} \int_0^{\hat{x}} z(x, y, 0) \; dx \; dy. \tag{3.9}$$

Taking m_0 as a reference, the normalized initial mass concentration of binder and nanoparticles are b_0/m_0 and z_0/m_0 respectively. Hence, the initial sum of partial volumes of the holographic grating can be expressed relative to m_0

$$\frac{v(0)}{m_0} = \frac{1}{\rho_b} \frac{b_0}{m_0} + \frac{1}{\rho_m} + \frac{1}{\rho_z} \frac{z_0}{m_0}. \tag{3.10}$$

For example, for the AA/PVA photopolymer with an 83:17 binder to monomer ratio doped 5 wt.% with MFI nanozeolites the initial volume is

$$\frac{v(0)}{m_0} = \frac{5.05}{1.19 \text{ g/cm}^3} + \frac{1}{1.15 \text{ g/cm}^3} + \frac{0.32}{1.78 \text{ g/cm}^3} = 5.29 \text{ cm}^3/\text{g}.$$

That is to say, although we cannot know exactly what mass of monomer is within the space of a single grating at $t = 0$, for every 1 g of AA monomer, the initial volume of that space is 5.29 cm^3. We can calculate the volume at time t if we have expressions for the total volume of monomer, short polymer, cross-linked polymer and nanoparticles inside the grating

$$v(t) = \frac{1}{\rho_m} \left[\frac{1}{\hat{x}T(t)} \int_0^{T(t)} \int_0^{\hat{x}} m \; dx \; dy \right] + \frac{1}{\rho_p} \left[\frac{1}{\hat{x}T(t)} \int_0^{T(t)} \int_0^{\hat{x}} p \; dx \; dy \right] +$$
$$\frac{1}{\rho_p} \left[\frac{1}{\hat{x}T(t)} \int_0^{T(t)} \int_0^{\hat{x}} q \; dx \; dy \right] + \frac{1}{\rho_z} \left[\frac{1}{\hat{x}T(t)} \int_0^{T(t)} \int_0^{\hat{x}} z \; dx \; dy \right] +$$
$$\frac{1}{\rho_b} \left[\frac{1}{\hat{x}T(t)} \int_0^{T(t)} \int_0^{\hat{x}} b \; dx \; dy \right]. \tag{3.11}$$

An important assumptions of the fringe-plane rotation model is that all loss of volume due to polymerization takes place in the thickness of the recording medium

$$\frac{T(t)}{T_0} = \frac{v(t)/m_0}{v(0)/m_0}. \tag{3.12}$$

Define nondimensional variables $\overline{x}$, $\overline{y}$, $\overline{t}$ and $u(t)$ such that

$$x = \hat{x}\overline{x}, \qquad y = T_0\overline{y}, \qquad t = t_0\overline{t}, \qquad T(t) = T_0 u(t),$$

$$\alpha_m^{(x)} = \frac{D_m t_0}{\hat{x}^2}, \qquad \alpha_m^{(y)} = \frac{D_m t_0}{T_0^2},$$

$$\alpha_p^{(x)} = \frac{D_p t_0}{\hat{x}^2}, \qquad \alpha_p^{(y)} = \frac{D_p t_0}{T_0^2}, \qquad \alpha_{pz}^{(x)} = \epsilon_{pz} z_0 \alpha_p^{(x)}, \qquad \alpha_{pz}^{(y)} = \epsilon_{pz} z_0 \alpha_p^{(y)},$$

$$\alpha_z^{(x)} = \frac{D_z t_0}{\hat{x}^2}, \qquad \alpha_z^{(y)} = \frac{D_z t_0}{T_0^2}, \qquad \alpha_{zq}^{(x)} = \epsilon_{qz} m_0 \alpha_z^{(x)}, \qquad \alpha_{zq}^{(y)} = \epsilon_{qz} m_0 \alpha_z^{(y)},$$

$$\alpha_{zp}^{(x)} = \epsilon_{pz} m_0 \alpha_z^{(x)}, \qquad \alpha_{zp}^{(y)} = \epsilon_{pz} m_0 \alpha_z^{(y)},$$

$$\xi^* = \xi z_0 \qquad \beta = F_0 t_0, \qquad \gamma = \Gamma m_0 t_0, \qquad \zeta^* = T_0 \zeta,$$

$$f^*(x, y, t) = e^{-a\zeta^*(u-y)} \left\{ 1 + V e^{-\xi^* z} \cos\left[2\pi \left(x - \frac{T_0}{\hat{x}} \tan\phi_r y \right) \right] \right\}.$$

The nondimensionalized mass concentration of monomer, short polymer, cross-linked polymer and inorganic nanoparticles within the holographic grating are solutions to

$$\frac{\partial m}{\partial t} = \alpha_m^{(x)} \frac{\partial^2 m}{\partial x^2} + \alpha_m^{(y)} \frac{\partial^2 m}{\partial y^2} - \Phi(t)\beta f^*(x, y, t) m, \tag{3.13a}$$

$$\frac{\partial p}{\partial t} = \alpha_p^{(x)} \frac{\partial^2 p}{\partial x^2} + \alpha_p^{(y)} \frac{\partial^2 p}{\partial y^2} + \alpha_{pz}^{(x)} \frac{\partial^2 (pz)}{\partial x^2}$$
$$+ \alpha_{pz}^{(y)} \frac{\partial^2 (pz)}{\partial y^2} + \Phi\beta f^*(x, Y, t) m - \Phi(t)\gamma p^2, \tag{3.13b}$$

$$\frac{\partial q}{\partial t} = \Phi(t)\gamma p^2, \tag{3.13c}$$

$$\frac{\partial z}{\partial t} = \alpha_z^{(x)} \frac{\partial^2 z}{\partial x^2} + \alpha_z^{(y)} \frac{\partial^2 z}{\partial y^2} + \alpha_{pz}^{(x)} \frac{\partial^2 (pz)}{\partial x^2}$$
$$+ \alpha_{pz}^{(y)} \frac{\partial^2 (pz)}{\partial y^2} + \alpha_{qz}^{(x)} \frac{\partial^2 (qz)}{\partial x^2} + \alpha_{qz}^{(y)} \frac{\partial^2 (qz)}{\partial y^2}. \tag{3.13d}$$

The nondimensionalized domain is

$$0 \le x \le 1, \qquad 0 \le y \le u(t), \qquad t \ge 0.$$

A numerical solution to problems such as this involving a time dependent boundary can be resolved via a technique known as boundary immobilization methods [78, 79, 80] whereby a suitable transformation will bring the governing equation into a fixed

domain. Applying the following transformations

$$Y = \frac{y}{u(t)}, \qquad m(x, y, t) = M(x, Y, t), \qquad p(x, y, t) = P(x, Y, t),$$

$$q(x, y, t) = Q(x, Y, t), \qquad z(x, y, t) = Z(x, Y, t),$$

to Eqn. 3.13 the system becomes

$$\frac{\partial M}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial M}{\partial Y} + \alpha_m^{(x)}\frac{\partial^2 m}{\partial x^2} + \alpha_m^{(y)}\frac{1}{u^2}\frac{\partial^2 M}{\partial Y^2} - \Phi(t)\beta F^*(x, Y, t)M, \qquad (3.14a)$$

$$\frac{\partial P}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial P}{\partial Y} + \alpha_p^{(x)}\frac{\partial^2 P}{\partial x^2} + \alpha_p^{(y)}\frac{1}{u^2}\frac{\partial^2 P}{\partial Y^2} + \alpha_{pz}^{(x)}\frac{\partial^2 (PZ)}{\partial x^2} +$$

$$\alpha_{pz}^{(y)}\frac{1}{u^2}\frac{\partial^2 (PZ)}{\partial Y^2} + \Phi\beta F^*(x, Y, t)M - \Phi(t)\gamma P^2, \qquad (3.14b)$$

$$\frac{\partial Q}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial Q}{\partial Y} + \Phi(t)\gamma P^2, \qquad (3.14c)$$

$$\frac{\partial Z}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial Z}{\partial Y} + \alpha_z^{(x)}\frac{\partial^2 Z}{\partial x^2} + \alpha_z^{(y)}\frac{1}{u^2}\frac{\partial^2 Z}{\partial Y^2} + \alpha_{pz}^{(x)}\frac{\partial^2 (PZ)}{\partial x^2} +$$

$$\frac{1}{u^2}\alpha_{pz}^{(y)}\frac{\partial^2 (PZ)}{\partial Y^2} + \alpha_{qz}^{(x)}\frac{\partial^2 (QZ)}{\partial x^2} + \alpha_{qz}^{(y)}\frac{1}{u^2}\frac{\partial^2 (QZ)}{\partial Y^2}, \qquad (3.14d)$$

where

$$F^*(x, Y, t) = e^{-a\zeta^* u(1-Y)}\left\{1 + Ve^{-\xi^* Z}\cos\left[2\pi\left(x - \frac{T_0}{\hat{x}}\tan\phi_r uY\right)\right]\right\}. \qquad (3.15)$$

There is an additional equation required for the binder which is constant and uniform throughout recording

$$\frac{\partial b}{\partial t} = 0.$$

The boundary immobilization must also be applied to the mass concentration of the binder, $b(x, y, t) = b_0\bar{b}(x, y, t) = b_0 B(x, Y, t)$, hence

$$\frac{\partial B}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial B}{\partial Y}.$$

Applying the boundary immobilization transformation to the initial (Eqn. 3.4) and boundary conditions (Eqn. 3.6) gives us

$$M(x, Y, 0) = 1, \qquad P(x, Y, 0) = 0, \qquad Q(x, Y, 0) = 0, \qquad Z(x, Y, 0) = 1,$$

$$B(x, Y, 0) = 1, \qquad u(0) = 1, \qquad u'(0) = 0,$$

$$\frac{\partial^n M}{\partial x^n}(0, Y, t) = \frac{\partial^n M}{\partial x^n}(1, Y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$

$$\frac{\partial^n P}{\partial x^n}(0, Y, t) = \frac{\partial^n P}{\partial x^n}(1, Y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$

$$\frac{\partial^n Z}{\partial x^n}(0, Y, t) = \frac{\partial^n Z}{\partial x^n}(1, Y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$

$$\frac{\partial M}{\partial Y}(x, 0, t) = \frac{\partial P}{\partial Y}(x, 0, t) = \frac{\partial Q}{\partial Y}(x, 0, t) = \frac{\partial Z}{\partial Y}(x, 0, t) = \frac{\partial B}{\partial Y}(x, 0, t) = 0,$$

$$\frac{\partial M}{\partial Y}(x, 1, t) = \frac{\partial P}{\partial Y}(x, 1, t) = \frac{\partial Q}{\partial Y}(x, 1, t) = \frac{\partial Z}{\partial Y}(x, 1, t) = \frac{\partial B}{\partial Y}(x, 1, t) = 0.$$

Applying the nondimensionalization and the boundary immobilization transformation to Eqn. 3.11, the volume can be expressed as

$$\frac{v(t)}{m_0} = \left[\int_0^1 \int_0^1 \frac{M}{\rho_m} + \frac{P}{\rho_p} + \frac{Q}{\rho_p} + \frac{z_0/m_0 Z}{\rho_z} + \frac{b_0/m_0}{\rho_b} \, dx \, dY \right], \qquad (3.16)$$

hence Eqn. 3.12 is reduced to,

$$u(t) = \left[\frac{1}{\rho_b} \frac{b_0}{m_0} + \frac{1}{\rho_m} + \frac{1}{\rho_z} \frac{z_0}{m_0} \right]^{-1} \left[\int_0^1 \int_0^1 \frac{M}{\rho_m} + \frac{P}{\rho_p} + \frac{Q}{\rho_p} + \frac{z_0/m_0 Z}{\rho_z} + \frac{b_0/m_0}{\rho_b} \, dx \, dY \right].$$
$$(3.17)$$

Polymerization induced shrinkage can be modelled as

$$\text{Volume Shrinkage} = \frac{u(0) - u(t)}{u(0)} = 1 - u(t). \qquad (3.18)$$

The holographic material is attached to a rigid substrate, this glass backing prevents any change in dimensions at the far side of the material. The x-direction between two successive peaks in the illumination pattern $\hat{x}$ must remain fixed throughout holographic recording and hence the Bragg planes need to rotate to a new slant angle, see Fig. 3.4. Using solutions of Eqn. 3.17 with Eqns. 3.1 and 3.2, for $u \neq 0$ and $0 < \phi_r < \pi/2$

$$\phi_r(t) = \tan^{-1} \left[\frac{\tan \phi_r(0)}{u(t)} \right], \qquad (3.19\text{a})$$

$$\Lambda(t) = \hat{x} \cos \phi_r(t), \qquad (3.19\text{b})$$

$$\hat{y}(t) = \frac{\Lambda(t)}{\sin \phi_r(t)}. \qquad (3.19\text{c})$$

Details of the numerical scheme for Eqn. 3.13 can be found in Appendix A. Numerical solutions of the above PDEs are presented in Figs. 3.8, 3.9, 3.10 and 3.11. The simulation is of a 50 μm thick hybrid AA/PVA photopolymer doped 5 wt.%

with MFI nanozeolites slanted at 10° and exposed to an interference pattern with intensity 1 mW/cm^2 and spatial frequency 1000 lines/mm for 50 s.

(a) At fixed surface ($Y = 0$)　　　　　　(b) At free surface ($Y = 1$)
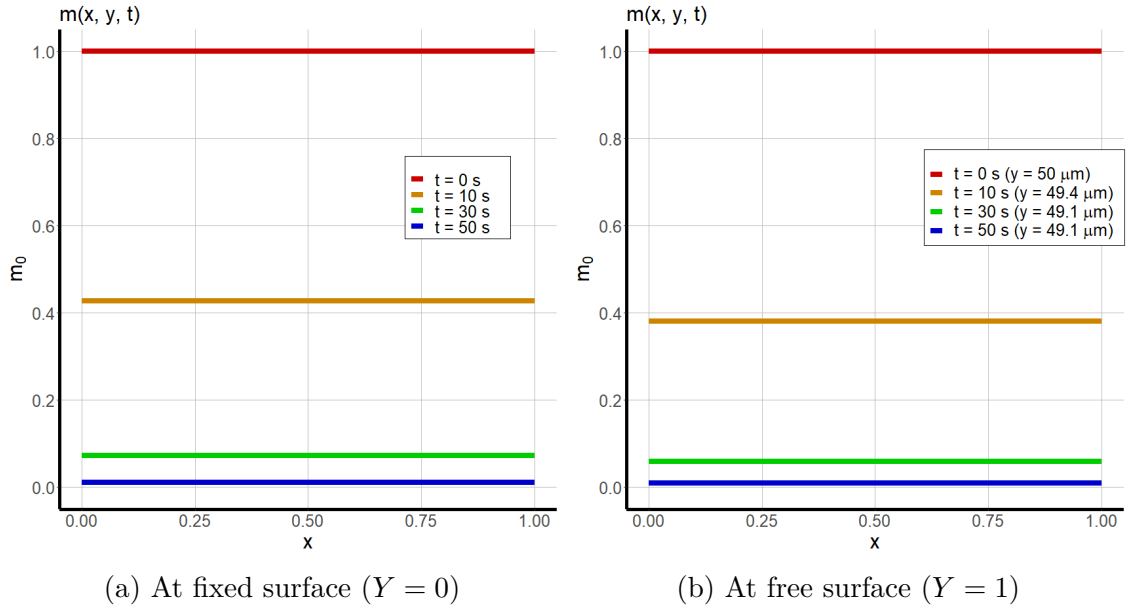
Fig. 3.8 Distribution of monomer at four different times at the fixed surface and at the time dependent free surface.

(a) At fixed surface ($Y = 0$)　　　　　　(b) At free surface ($Y = 1$)
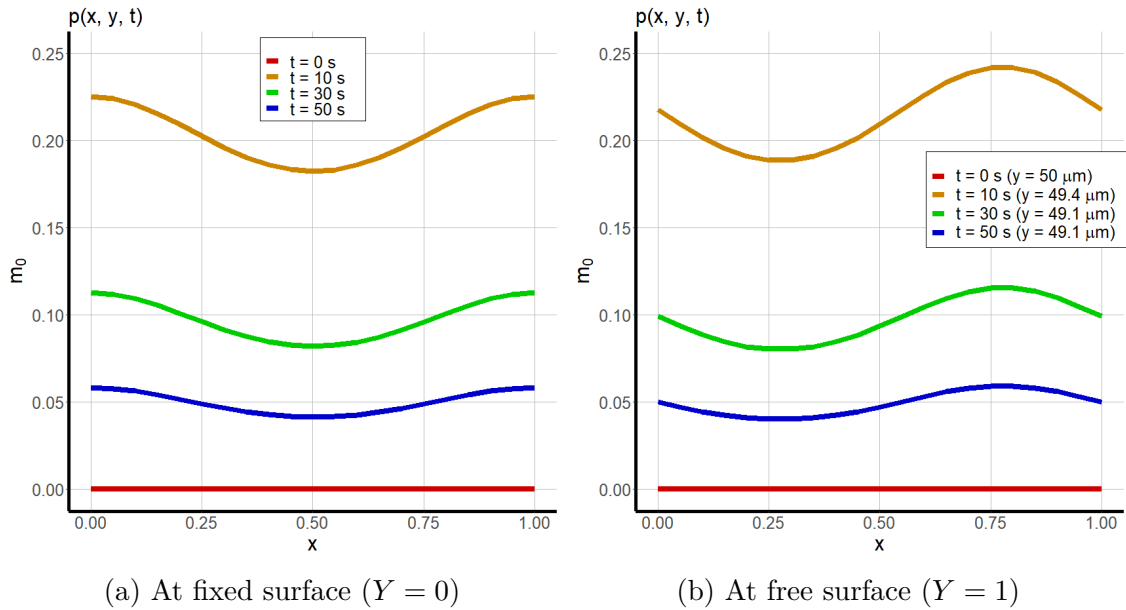
Fig. 3.9 Distribution of short polymer at four different times at the fixed surface and at the time dependent free surface.

(a) At fixed surface ($Y = 0$) (b) At free surface ($Y = 1$)
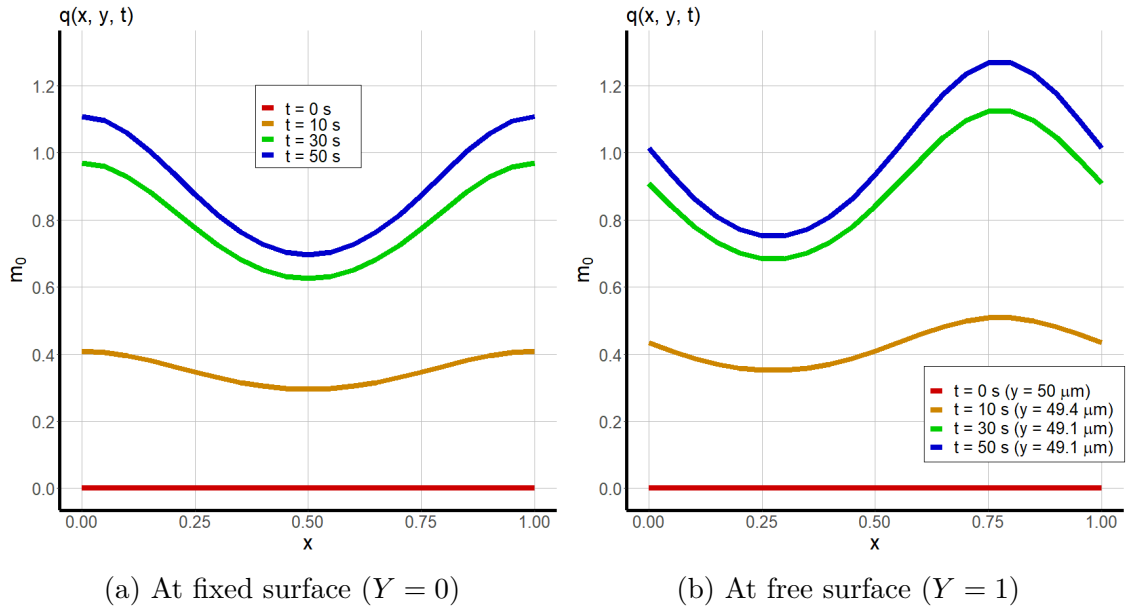
Fig. 3.10 Distribution of immobile polymer at four different times at the fixed surface and at the time dependent free surface.

(a) At fixed surface ($Y = 0$) (b) At free surface ($Y = 1$)
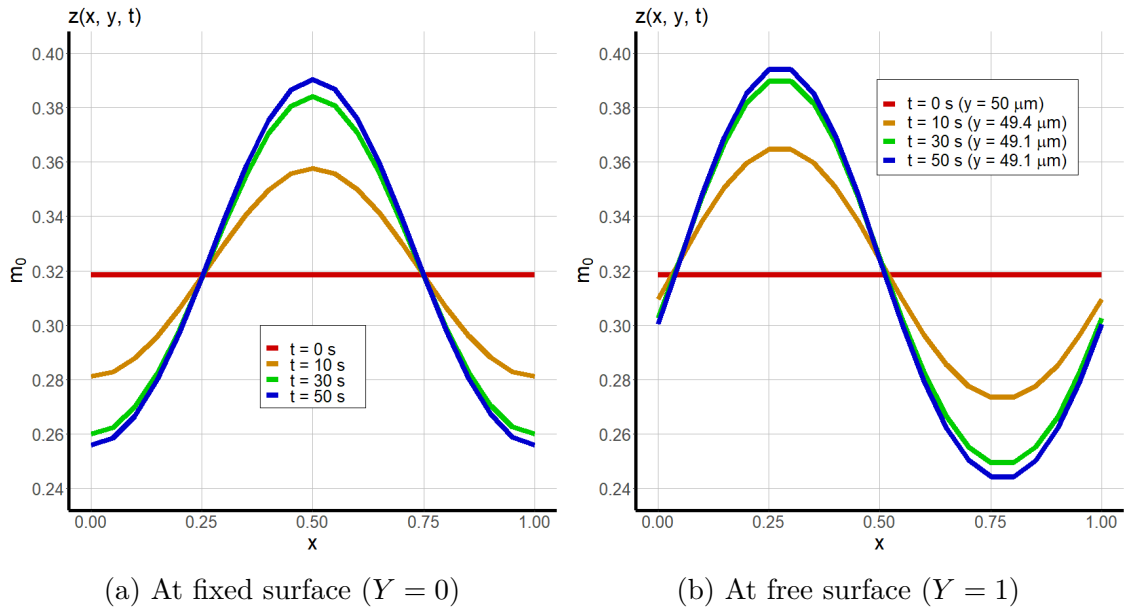
Fig. 3.11 Distribution of nanoparticles at four different times at the fixed surface and at the time dependent free surface.

The model predictions show that at the end of holographic recording virtually all monomer has been consumed and short polymer has been immobilized. Due to the use of a non-zero absorption coefficient ($\zeta = 139$ cm^{-1}) the model predicts a greater density of immobilized polymer chains near the surface which in turn leads to cross-diffusion of nanoparticles in the y-direction and results in a greater density of nanoparticles at the fixed surface of the material ($Y = 0$). Notice in Figs. 3.10 and 3.11 that the non-zero slant angle results in a phase difference between the fixed and free surface. The rotation of Bragg planes due to polymerization induced shrinkage means that there will be a new reconstruction geometry for maximum diffraction

efficiency, in other words, a detuning of the Bragg angle. The rotation of the slant angle at time t is

$$\Delta \phi_r = \phi_r(0) - \phi_r(t). \tag{3.20}$$

The predicted rotation of the Bragg planes is plotted against time for $\rho_p - \rho_m = 0, 0.05, 0.1$ and 0.15 g/cm^3; for $b_0/m_0 = 2, 3, 4$ and 5; for $\zeta = 0, 139$ and 277 cm^{-1}; and for recording intensities $1, 5$ and 10 mW/cm^2 in Fig. 3.12. The values of ζ are such that the recording intensity at the fixed surface of a $50\mu m$ sample is $100\%, 50\%$ and 25% respectively of the free surface intensity.

(a) Density Difference   (b) Binder

(c) Absorption   (d) Recording Intensity
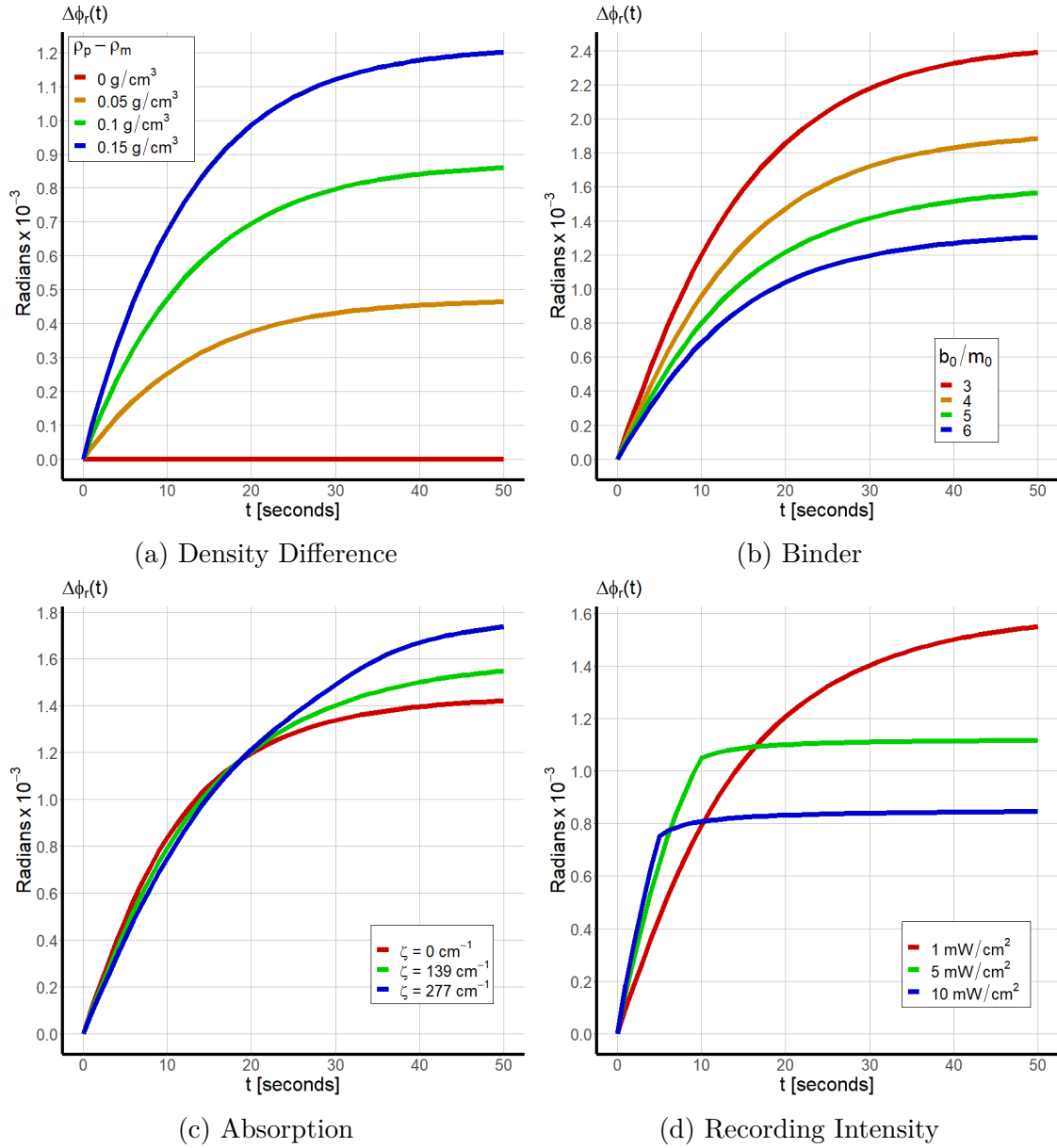
Fig. 3.12 Predicted rotation of slant angle.

Figs. 3.12a & 3.12b show that, as expected, a greater difference between the density of polymer and monomer results in greater shrinkage and that there is less

shrinkage when the binder makes up a larger part of the initial volume. Fig. 3.12c shows that polymerization induced shrinkage has a weak dependence on absorption. By increasing ζ, there is less polymerization taking place at the fixed surface ($Y = 0$) and monomer must move a greater distance to reach a point of initiation. In each of the three numerical simulations presented in Fig. 3.12d the duration of recording is varied so that energy input is fixed at 50 mJ/cm^2. Numerical simulation of the model predicts that shrinkage is minimized in high intensity recording. This is because under these conditions polymerization takes place on a much shorter time scale than the diffusion of monomer, the recording ends before the concentration of monomer near the point of initiation can be restocked. For a more complete understanding of shrinkage in holographic gratings, one must consider the optical properties.

3.3 RI & Shrinkage in Slanted Holographic Gratings

Using the same technique as in the previous chapters, the RI as a function of x, y and t is calculated via solutions to the coupled PDEs (Eqn. 3.13) and the Lorentz-Lorenz equation. The predicted evolution of the RI profile over the duration of the simulated holographic recording mentioned above is presented in Fig. 3.13.

(a) At fixed surface ($Y = 0$)   (b) At free surface ($Y = 1$)
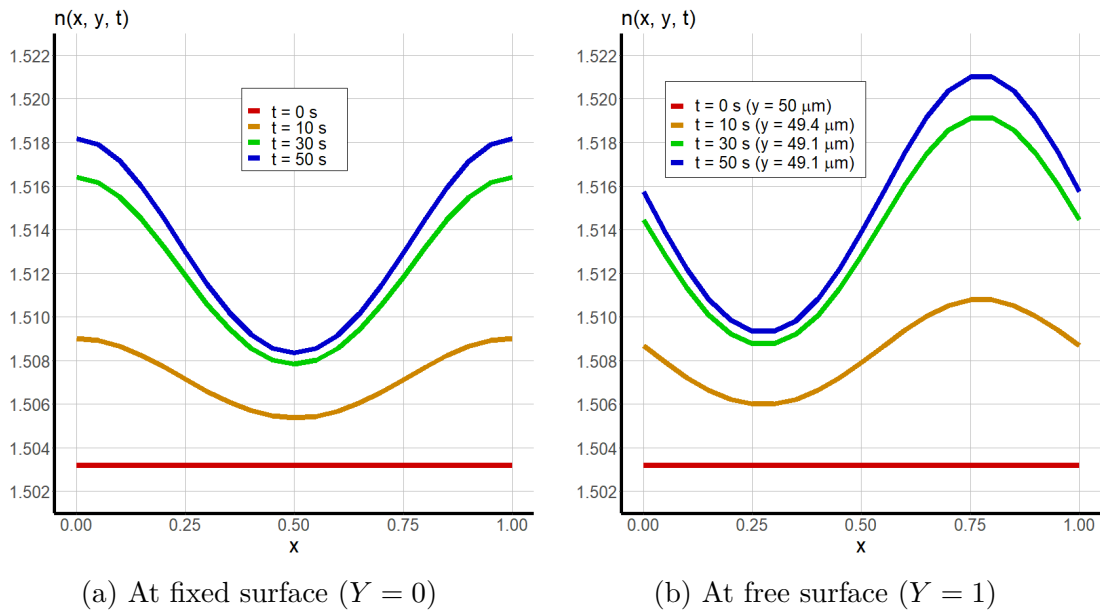
Fig. 3.13 RI profile at four different times at the fixed surface and at the time dependent free surface.

The mean RI inside the holographic grating can be modelled by

$$\overline{n}(t) = \frac{1}{\hat{x}T} \int_0^T \int_0^{\hat{x}} n(x,y,t) \, dx \, dy = \int_0^1 \int_0^1 n(x,Y,t) \, dx \, dY. \qquad (3.21)$$

The predicted change in mean RI inside the holographic grating for nanoparticle doping ranging from 0-5 wt.% and nanoparticle RI ranging from 1.366 to 2.1 is plotted against time in Fig. 3.14.

(a) Nanoparticle Doping   (b) Nanoparticle RI
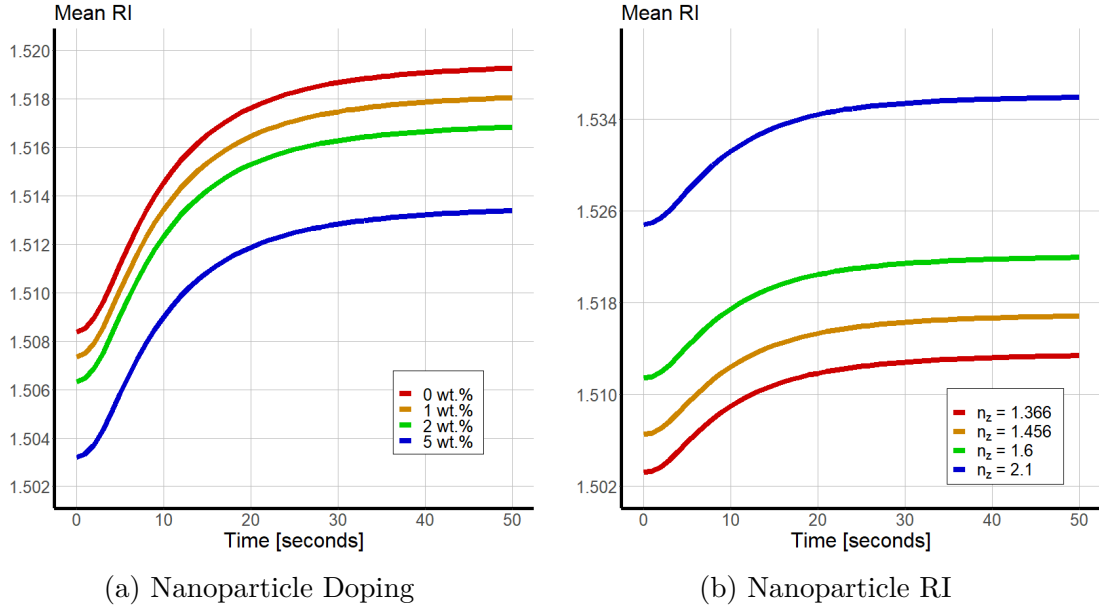
Fig. 3.14 The change in mean RI within the holographic gratings recorded in hybrid photopolymer media.

In experimental studies, polymerization induced shrinkage is calculated by measuring the detuning of the Bragg angle. But the changes to the RI and grating period will result in an additional detuning, from Eqn. 1.2

$$\theta_B(t) = \sin^{-1}\left(\frac{\lambda_r}{2\overline{n}(t)\Lambda(t)}\right) - \phi_r(t).$$

The total Bragg detuning resulting from both rotation of Bragg planes and the changes to RI and grating period is $\Delta\theta_B = \theta_B(0) - \theta_B(t)$. Consequently, an apparent shrinkage will be observed in experimental studies; Eqn. 1.16 becomes

$$\text{Apparent Shrinkage} = 1 - \frac{\tan\phi_r(0)}{\tan[\phi_r(0) + \Delta\theta_B]}. \tag{3.22}$$

There are a variety of different parameters which can be investigated numerically to see if they influence shrinkage. These parameters can be grouped into those relating to the inorganic nanoparticles (wt.%, n_z and ρ_z); those relating to the binder (b_0 and n_b); those relating to the polymer ($\rho_p - \rho_m$ and n_q); those relating to the recording conditions (spatial frequency Λ, recording intensity I_0 and slant angle ϕ); and those relating to the sample (initial thickness T_0 and absorption ζ). The predicted final actual shrinkage (Eqn. 3.18) and apparent shrinkage (Eqn. 3.22) is plotted against a range of different parameters in Figs. 3.15, 3.16, 3.17, 3.18 and 3.19.

(a) Doping

(b) Nanoparticle RI

(c) Nanoparticle density
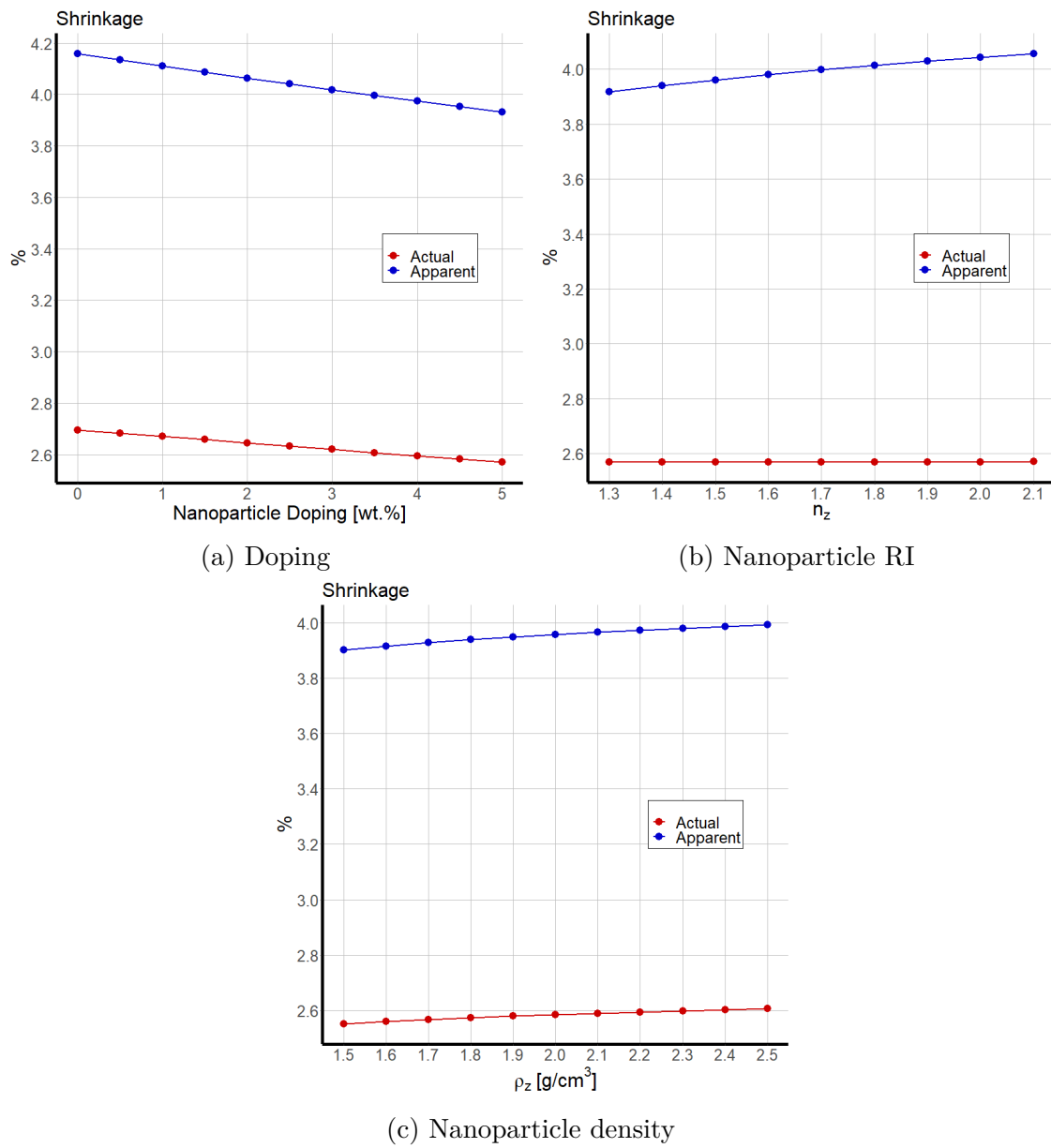
Fig. 3.15 The predicted actual and apparent shrinkage for a variety of different parameters relating to inorganic nanoparticles.

(a) Binder RI
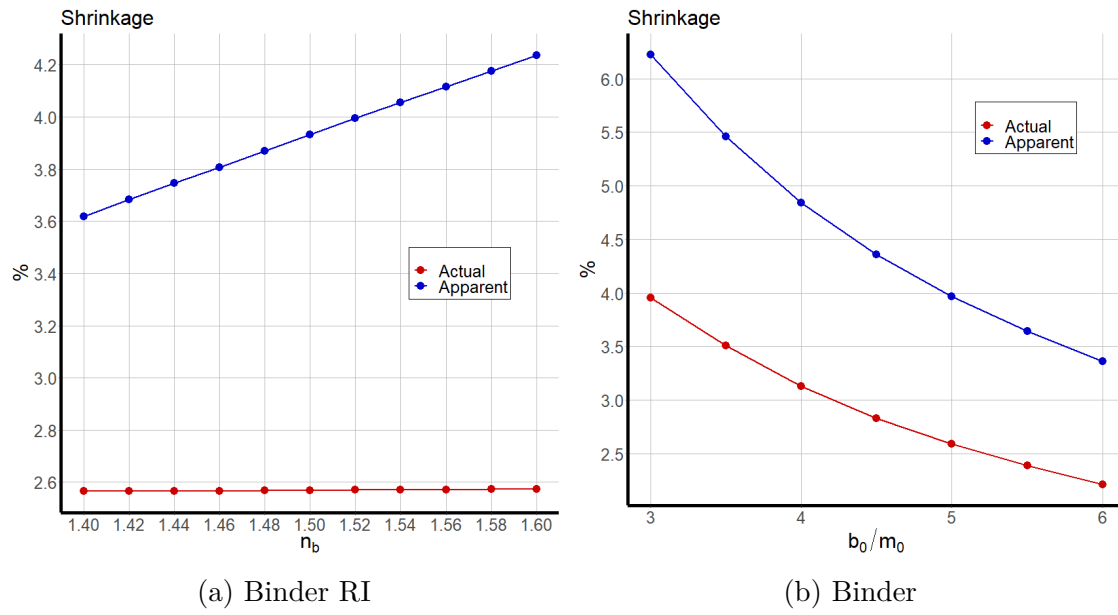
(b) Binder

Fig. 3.16 The predicted actual and apparent shrinkage for a variety of different parameters relating to the binder.

(a) Difference in density

(b) Polymer RI
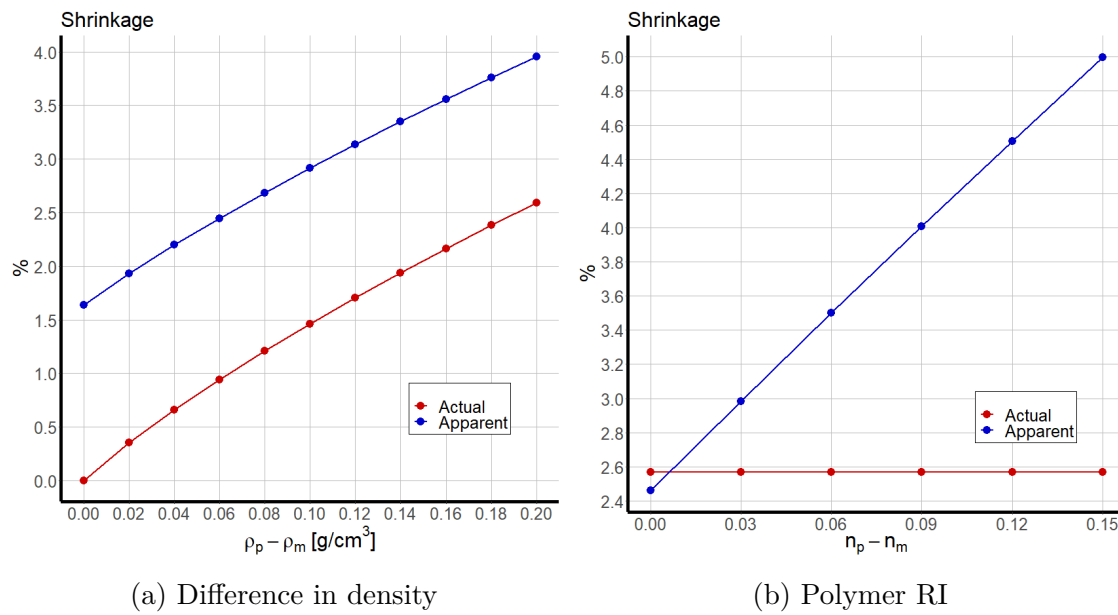
Fig. 3.17 The predicted actual and apparent shrinkage for a variety of different parameters relating to the polymer.

(a) Spatial frequency

(b) Recording Intensity

(c) Slant angle
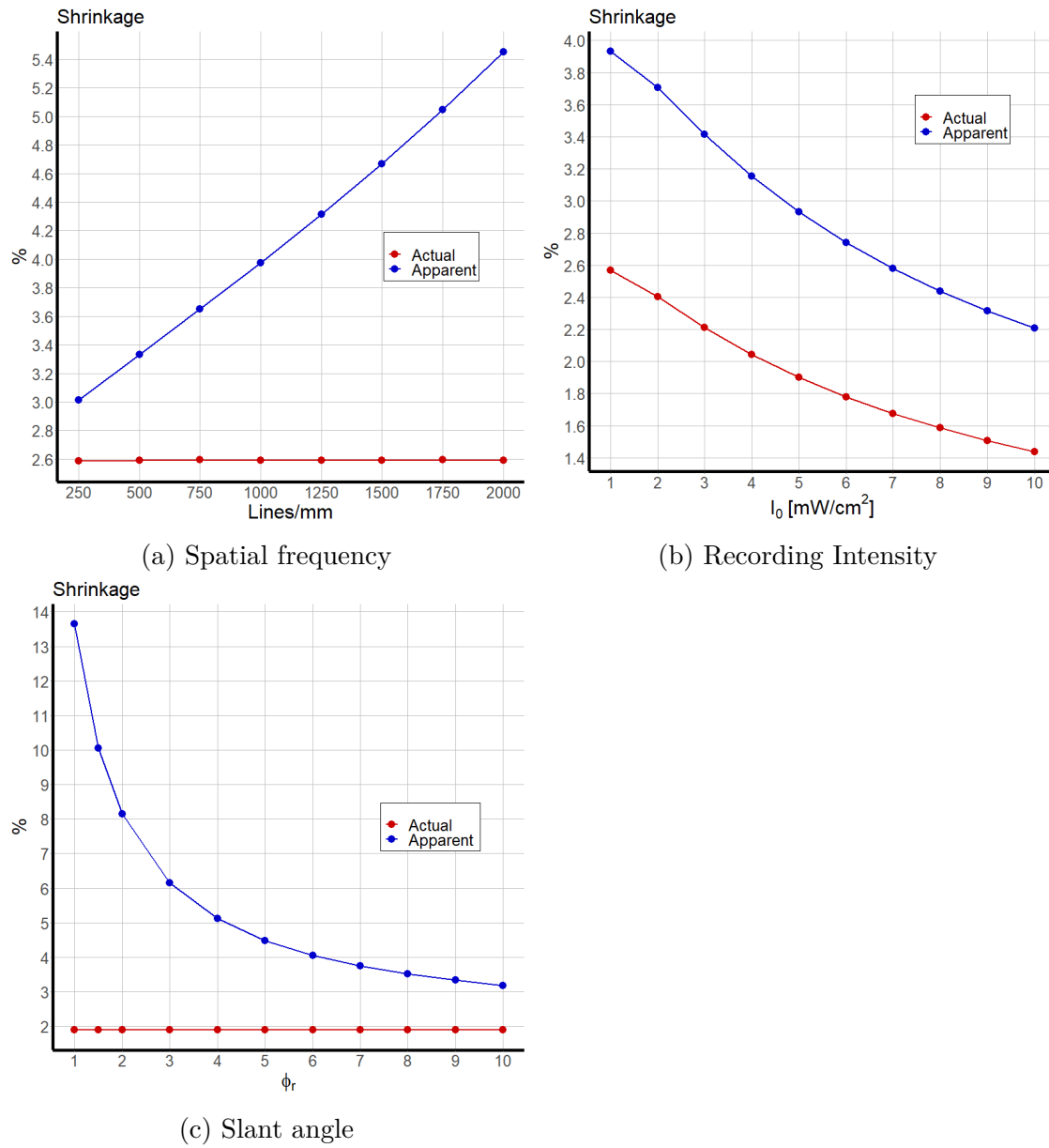
Fig. 3.18 The predicted actual and apparent shrinkage for a variety of different parameters relating to the recording conditions.

(a) Initial thickness   (b) Absorption
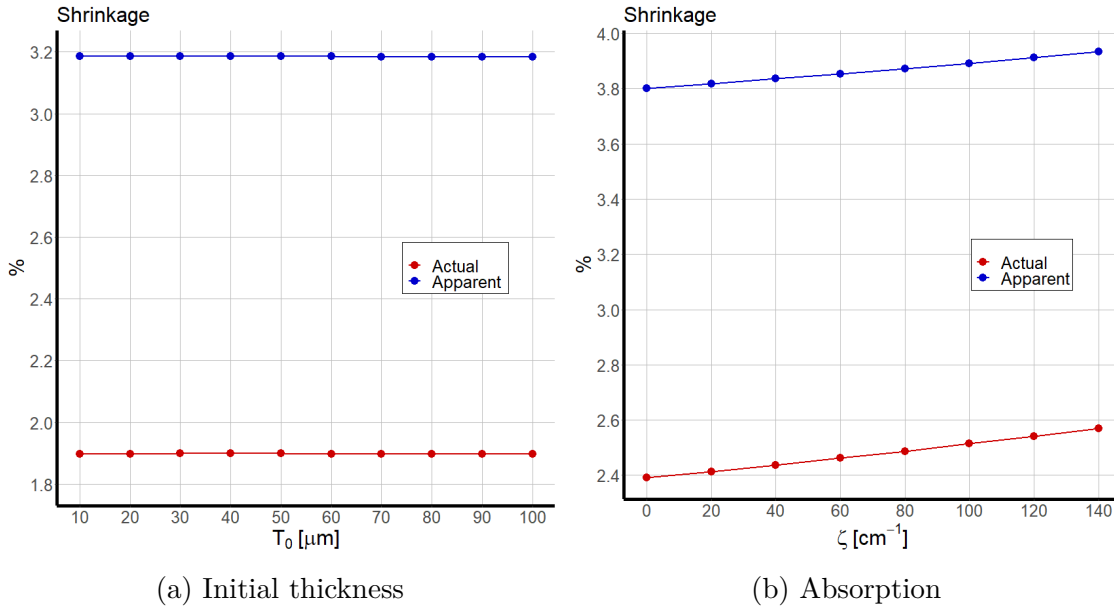
Fig. 3.19 The predicted actual and apparent shrinkage for a variety of different parameters relating to the sample.

The most noteworthy predictions are those found in Figs. 3.15a and 3.19a. The model found a small drop in shrinkage with increased nanoparticle doping. Although the overlying trend is consistent with the findings from [75, 59], which reported shrinkage of 1.3% in undoped samples and dropped to 0.9% in samples doped 2.5 wt.% with nanozeolites; 0.7% in samples doped 5 wt.%; and to 0.6% in samples doped 7.5 wt.%, the quantitative extent of the drop is insufficient compared with the observed values. The predicted shrinkage was also unaffected by the initial thickness, again, this does not agree with the findings of [72] which found shrinkage of 1.58% in samples with thickness of 30μm, dropping to 1.08% in samples 60μm thick and 0.58% in samples 120μm thick. However, Fig. 3.18a shows a significant increase in apparent shrinkage at high spatial frequencies, this supports the findings of [72, 4]. The variables which significantly effect the model's predicted actual shrinkage are the portion of monomer at time t that has been polymerized (see Fig. 3.12); the relative quantity of binder in the initial resin; the difference in density between polymer and monomer; the recording intensity (for a fixed energy input); and (to a small extent) the absorption coefficient. The RI of binder, polymer and (again to a small extent) the nanoparticle makes a significant difference to apparent shrinkage. Spatial frequency and slant angle also have a significant influence on apparent shrinkage.

3.4   RI Modulation

The RI is still a periodic function, $n(x, y, t) = n(x+\hat{x}, y, t)$, and can still be represented by a Fourier series with an adjustment to accommodate how the phase changes with depth.

$$n(x, y, t) \approx \sum_{i=0} A_i(y, t) \cos\left(\frac{2\pi}{\Lambda} ix\right) + B_i(y, t) \sin\left(\frac{2\pi}{\Lambda} ix\right),$$

$$A_0(y,t) = \frac{1}{\Lambda} \int_0^\Lambda n(x,y,t) \; dx, \tag{3.23}$$

$$A_1(y,t) = \frac{2}{\Lambda} \int_0^\Lambda n(x,y,t) \cos\left(\frac{2\pi}{\Lambda}x\right) \; dx, \tag{3.24}$$

$$B_1(y,t) = \frac{2}{\Lambda} \int_0^\Lambda n(x,y,t) \sin\left(\frac{2\pi}{\Lambda}x\right) \; dx. \tag{3.25}$$

RI modulation can be modelled as

$$\Delta n(y,t) = 2\sqrt{A_1^2 + B_1^2}. \tag{3.26}$$

The predicted time evolution of the RI modulation in the recording of a 10° slanted grating at $Y = 0, 0.5$ and 1 for zero and non-zero absorption are presented in Fig. 3.20

(a) $\zeta = 0$ cm^{-1} (b) $\zeta = 139$ cm^{-1}
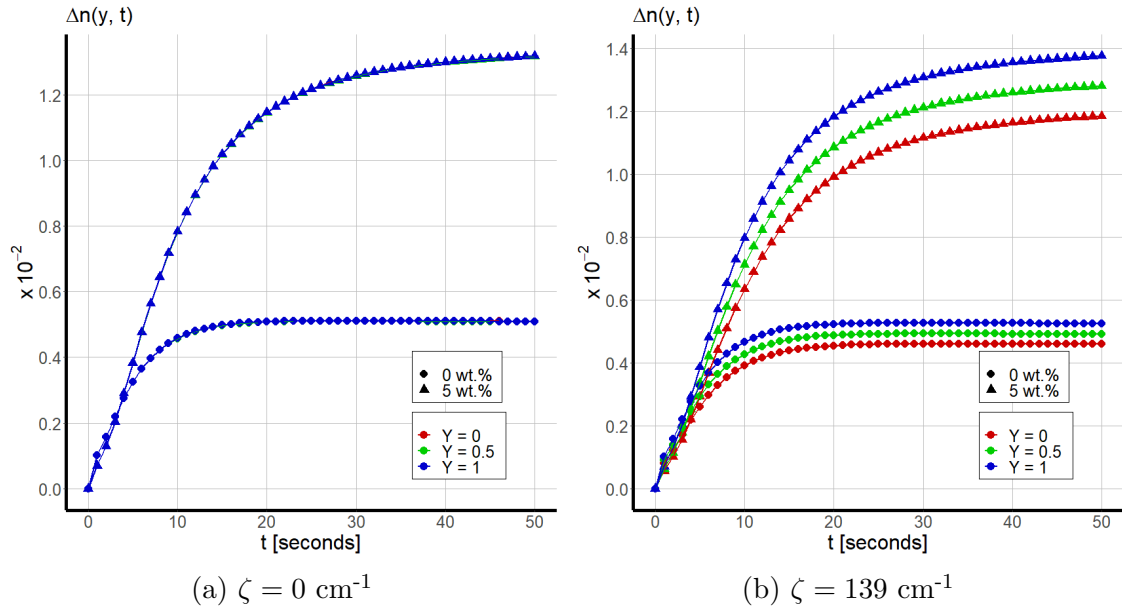
Fig. 3.20 RI modulation in 10° slanted holographic gratings at three different depths.

See that for zero absorption there is so little variation in RI modulation across the depth of the holographic grating recorded in both hybrid and undoped photopolymer media that the three different curves corresponding to $Y =0$, 0.5 and 1 cannot be distinguished. In the non-zero absorption simulation there is greater polymerization near the free surface ($Y = 1$), hence there is greater polymer growth and Δn at the free surface than at the fixed surface ($Y = 0$). The predicted variation in Δn across the surfaces also increases with increased thickness. The predicted RI modulation over a 250 mJ/cm^2 recording in undoped and hybrid photopolymer samples with initial thickness ranging from 10-90 μm, slant angles -10°, 0° and $+10^o$ and $\zeta = 139$ cm^{-1} is presented in Figs. 3.21 and 3.22.

(a) 0 wt.%   (b) 5 wt.%
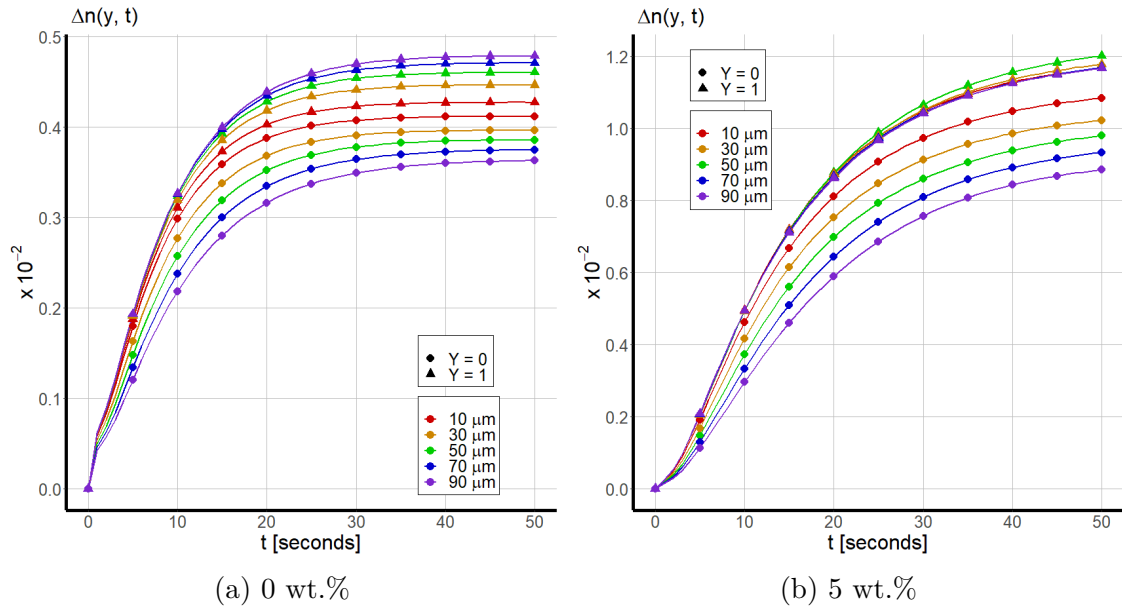
Fig. 3.21 RI modulation in 10º slanted holographic gratings for initial thickness ranging from 10-90 μm.

In both undoped and hybrid photopolymer samples the variation in Δn across the surfaces increases with increased thickness. As one should expect, increasing the initial thickness will increase the RI modulation at the free surface where polymerization is at its maximum and reduce RI modulation at the fixed surface. However, the slant angle does not effect Δn, see Fig. 3.22.

(a) 0 wt.%   (b) 5 wt.%
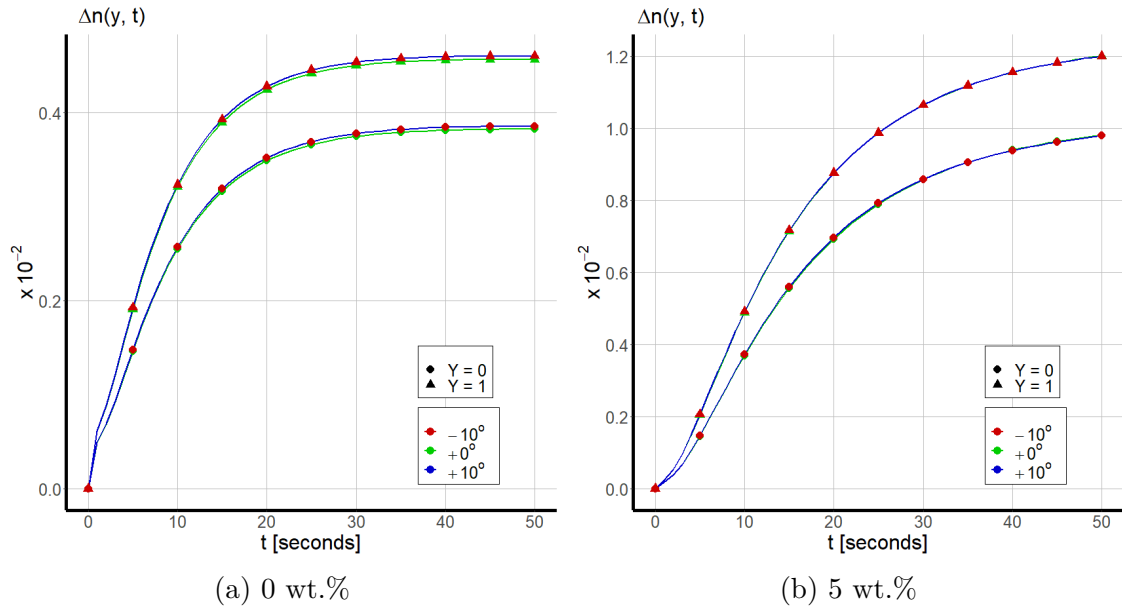
Fig. 3.22 RI modulation in 50 μm thick slanted holographic gratings for $\phi = -10º$, 0º and +10º.

3.5 Distortion

Deviation of the theoretically modelled holographic gratings from the recording pattern is measured by distortion δ defined as

$$\delta(y,t) = \left\{ \frac{1}{\Lambda} \int_0^\Lambda [n(x,y,t)]^2 dx \right\}^{-1} \frac{1}{\Lambda} \int_0^\Lambda \left\{ n(x,y,t) - \tilde{n}(x,y,t) \right\}^2 \ dx, \qquad (3.27)$$

where $\tilde{n}(x,y,t)$ is equal to the sum of the zero and first-harmonic terms of the Fourier series approximation

$$\tilde{n}(x,y,t) = A_0(y,t) + A_1(y,t)\cos\left(\frac{2\pi}{\Lambda}x\right) + B_1(y,t)\sin\left(\frac{2\pi}{\Lambda}x\right).$$

As with RI modulation the distortion is a function of both depth y and time t. The predicted distortion in a slanted holographic gratings recorded in hybrid photopolymer media with thickness ranging from 10-90 μm and slant angles -10º, 0º and +10º is presented in Figs. 3.23 and 3.24.

(a) 0 wt.% (b) 5 wt.%
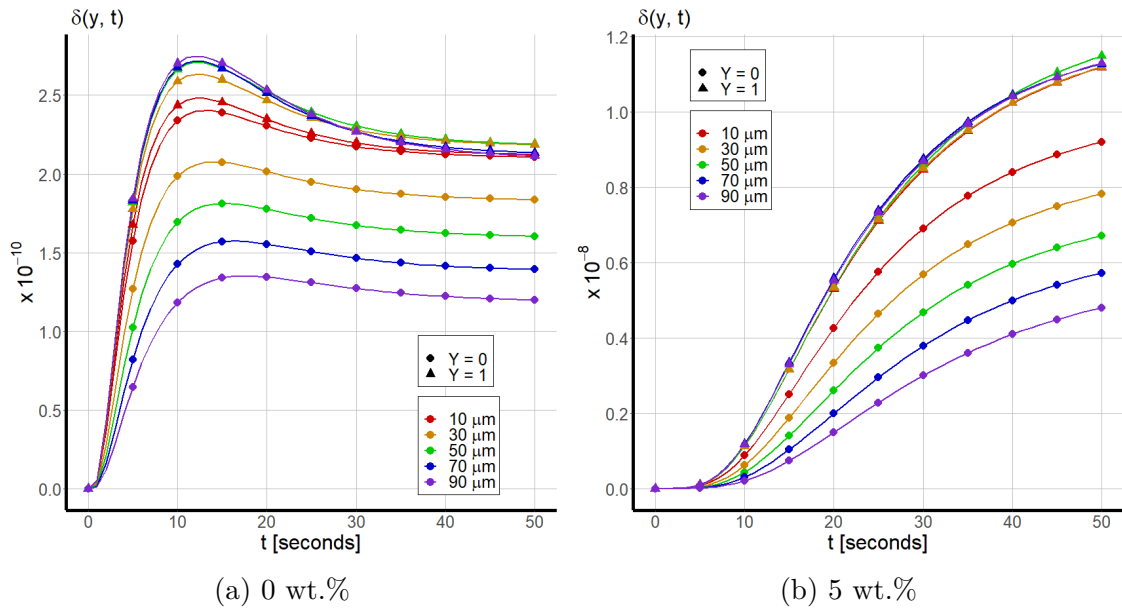
Fig. 3.23 Distortion in 10º slanted holographic gratings for initial thickness ranging from 10-90 μm.

(a) 0 wt.%   (b) 5 wt.%
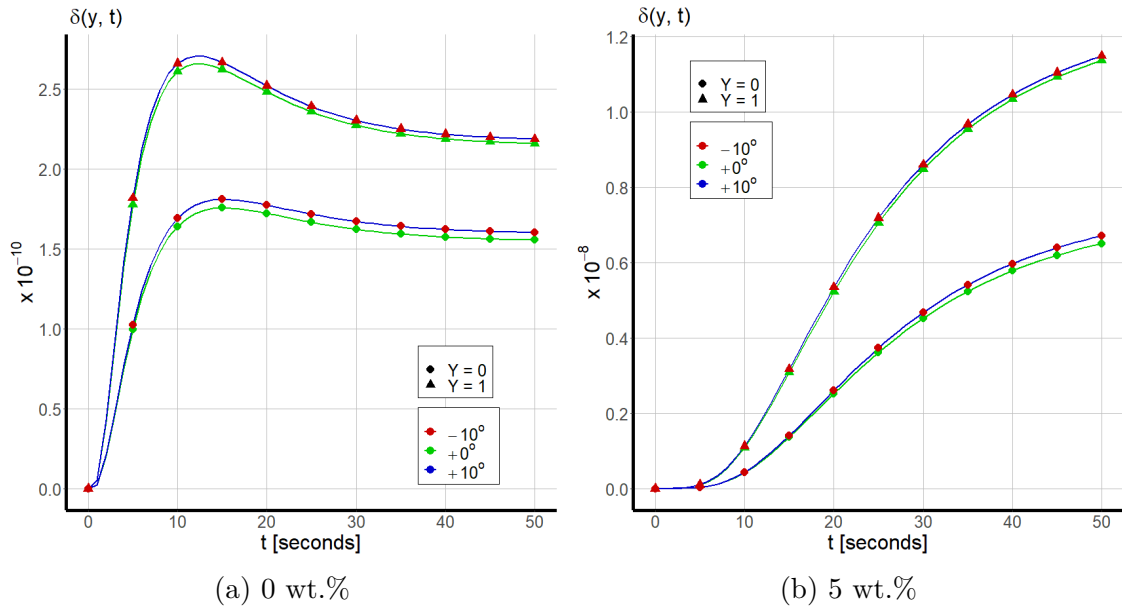
Fig. 3.24 Distortion in 50 μm thick slanted holographic gratings for $\phi = -10^{\circ}$, 0° and $+10^{\circ}$.

The model predicts significantly greater distortion in hybrid photopolymer media compared with undoped photopolymers with greater variation across the fixed and free surfaces increasing with increased initial thickness. Distortion is almost independent of the slant angle, the difference between the results for $+10^{\circ}$ and -10° is so small that the red and blue marker cannot be distinguished in Fig. 3.24.

3.6   Summary & Conclusions

Can the formation of slanted holographic gratings be modelled mathematically such that published experimental results, specifically the reduced shrinkage with increased nanoparticle doping [59] and increased shrinkage at high spatial frequencies [72], can be predicted theoretically?

This is the research question that this chapter aimed to address. The numerical simulations of the mathematical model described in this chapter have shown mixed success in predicting the findings from [75, 59, 72, 4]. However, the model has been somewhat successful in offering some insight into the formation of slanted holographic gratings in hybrid photopolymer media and how they may be optimized for the purpose of environmental sensing.

The existing equations described in Chapter 2 were insufficient for modelling polymerization induced shrinkage and it was deemed necessary to extend the equations to a second spatial dimension to accommodate the non-zero slant angle and absorption of light within the nanocomposite and the finite depth of the photonic structure. By extending the diffusion flux expressions to a direction parallel to the depth of the sample the temporal evolution and spatial redistribution of monomer, short polymer, cross-linked polymer and inorganic nanoparticles can be described by

a system of reaction-diffusion equations (Eqn. 3.14). The time dependent spatial domain required the use of boundary immobilization methods and a new polymerization function (Eqn. 3.15) was needed for the Beer-Lambert law and the shifting illumination pattern within the depth of the nanocomposite. By averaging the mass concentrations over the spatial domain it was shown how the volume of the grating could be expressed as a function of time. These expressions can be used to model polymerization induced shrinkage and the Bragg angle detuning. There is a change in the average RI and a change in the grating period inside the nanocomposite as a result of polymerization, consequently there is an additional Bragg angle detuning resulting in an apparent shrinkage (Eqn. 3.22) which will differ from the actual shrinkage (Eqn. 3.18). Although the results in Fig. 3.18a show there was success in predicting the observed increased shrinkage at high spatial frequencies the results in Fig. 3.15a and 3.19a show that there was very little success in predicting the fall in shrinkage with increased doping of inorganic nanoparticles and increased sample thickness that has been observed experimentally. However, the model has been able to offer some insight into which parameters do influence the final actual and apparent shrinkage. The actual shrinkage is largely determined by the mass ratio of binder to monomer; the difference in density between the polymer and monomer; the recording intensity (for a fixed energy input); the absorption coefficient; and the fraction of the initial quantity of monomer that has been polymerized. The apparent shrinkage is determined by the RI of the polymer, the binder and, to a lesser extent, the RI of the nanoparticles. The spatial frequency and slant angle also have a substantial impact on apparent shrinkage.

The RI of a slanted holographic grating can be modelled by a Fourier series with the first order harmonics acting as a good approximation for the RI modulation (Eqn. 3.26); the higher order harmonics are a measure of how the theoretically modelled holographic grating deviates from the illumination pattern (Eqn. 3.27). Numerical simulation of the model found that RI modulation is at its maximum within the interior of the photonic structure. By increasing the initial thickness of the sample the model predicted increased RI modulation at the free surface and reduced RI modulation at the fixed surface. The model also predicted lower distortion at the fixed surface and greater distortion at the free surface with increased thickness. The initial slant angle had no effect on the neither the predicted RI modulation nor distortion.

Now that we have established a mathematical model for the formation of slanted holographic gratings recorded in hybrid photopolymer media we can move on to the final objective of this research.

Chapter 4

Diffusion of Target Analytes in Holographic Sensors

As mentioned in Chapter 1, the interference of two coherent optical wavefronts produces an illumination pattern which can be stored as a permanent record in a photosensitive material (a hologram) and can be reconstructed by illuminating the hologram with the reference wave. To produce a perfect reconstruction both the reference wave and the hologram must be in the same condition as in the recording stage, a deviation in the physical dimensions or optical properties of the hologram from recording will result in a distortion of the reconstructed wavefront. This phenomenon is considered a caveat for some holographic applications such as data storage or holographic displays but holographic sensors exploit this feature of hologram reconstruction for the purpose of detecting physical changes in the local environment or the presence of a target analyte. Holographic sensors are a portable and cost efficient method of environmental monitoring, by measuring changes in the diffractive efficiency, the angular response or wavelength response the device can make a quantitative measurement of pollutants in a water sample; drugs or pathogens in the bloodstream; temperature, pressure or humidity of the local atmosphere. The observed change in the hologram can be either a change in the physical dimensions, i.e. shrinkage or swelling of the hologram layer volume, or a change in the optical properties (RI modulation or average RI). Hybrid photopolymer nanocomposites are strong candidates for sensor technologies as they offer greater molecular sensitivity. Porous nanozeolites, in particular, can be synthesized in such a way as to fine tune the pore shape and size such that the probability of a specific target analyte will permeate through the nanocomposite is high. The final phase of this research will answer the second primary research question this project was designed to investigate: *how can we use the host photopolymer material properties, the recording conditions and the nanodopant properties to control the final grating in a hybrid photopolymer structure and hence optimize its functionality for environmental sensing?* Two key objectives in the design of photonic devices for environmental sensing are:

1. To identify the RI modulation that will result in the greatest sensitivity of the structure to changes in material composition or environmental conditions; in other words, the greatest change in diffraction efficiency η in the presence of a target analyte.

2. High dimensional stability, recorded holograms can be distorted as a consequence of shrinkage during polymerization.

The equations developed throughout this research have focused on describing what happens in the recording process. The focus of this chapter will move towards modelling what happens after recording. Specifically, the diffusion of target analytes within the interior of the hologram and investigating how the depth of the sample effects the angular or wavelength response. The equations introduced in Chapters 2 and 3 will be used to create the initial conditions for new equations to model the response of a theoretically modelled holographic grating exposed to a target analyte and will illustrate how both design objectives can be achieved. One of the main questions for researchers interested in holographic sensors concerns the compatibility of analytes with the host material and which properties are influenced by which analytes. For example, zeolite beta film have proved highly effective as humidity sensors and have demonstrated high sensitivity, good reversibility and long life for low water concentration applications [81].

4.1　Occupation of Porous Nanoparticles

Theoretical modelling of the diffusion of target analytes through a holographic grating begins by making a number of assumptions. In the most simple model described in this work the following assumptions are made: 1) the mobile analyte is a molecule sufficiently small that it is miscible with the photopolymer, the polymer chains do not impede the diffusion of the analyte and their distribution will not influence the distribution of analyte within the photonic structure; 2) the nanocomposite and analyte are maintained at a constant temperature so there is no thermal swelling or shrinkage in the thickness of the structure; 3) the overall mass density of the photopolymer-analyte system is constant; 4) the system is in mechanical equilibrium, there are no unbalanced external forces; 5) there is no bulk flow of the analyte fluid, in other words the mass movement of analyte is driven by concentration gradients rather than by pressure gradients. There are two processes which must be considered: the diffusion of analyte through the holographic grating and the occupation of the previously vacant porous nanoparticles. Consider first the diffusion of deionized water at a constant temperature with a fixed concentration of target analyte dissolved. Molecules with a van der Waals radius of smaller than the pore radius of nanozeolites $(3.8 - 6.6$ Å [6]) can easily infiltrate the nanoparticle and permeate the photonic structure, this can include the water molecules (1.93 Å) which form the solvent that the target analyte molecules are dissolved in. There will be a change in the optical

properties of the holographic grating due to the difference between the RI of occupied nanozeolites and the RI of empty nanozeolites. The change in optical properties can be modelled by first defining the mass concentration of three new constituents; the analyte to which the holographic grating is exposed, s; vacant nanozeolites, $z^{(e)}$; and nanozeolites occupied with analyte molecules, $z^{(s)}$. The analyte is mobile, its mass transport is driven solely by its own concentration gradient with diffusion coefficient D_s. The formation of $z^{(s)}$ occurs through the interaction of empty nanozeolites with the analyte, the constant γ_s represents the rate of capture. Once analyte molecules have occupied the porous nanozeolites they can either tunnel through the photonic structure via moving from the interior of one nanoparticle to that of an adjacent nanoparticle or the molecules can escape from the nanozeolite into the polymer matrix leaving behind it a now vacant nanozeolite, the rate of escape is measured by the constant ω_s.

As a starting point, we can model the very simple case of an unslanted holographic grating exposed to an analyte. The formation of an unslanted grating can be modelled using the equations outlined in Chapter 2. For

$$0 \leq x \leq \Lambda \qquad \text{and} \qquad t \geq 0,$$

the occupation of porous nanoparticles with analyte molecules in a theoretically modelled nanocomposite is governed by

$$\frac{\partial s}{\partial t} = D_s \frac{\partial^2 s}{\partial x^2} - \gamma_s s z^{(e)} + \omega_s z^{(s)}, \tag{4.1a}$$

$$\frac{\partial z^{(e)}}{\partial t} = -\gamma_s s z^{(e)} + \omega_s z^{(s)}, \tag{4.1b}$$

$$\frac{\partial z^{(s)}}{\partial t} = +\gamma_s s z^{(e)} - \omega_s z^{(s)}. \tag{4.1c}$$

We assume that the diffusion of analyte across the depth of the photonic structure takes place on a significantly shorter time scale compared with the capture of analyte so that the initial conditions are

$$s(x, 0) = s_0, \qquad z^{(s)}(x, 0) = 0, \qquad z^{(e)}(x, 0) = z(x),$$

where $z(x)$ is the concentration profile of nanozeolites at the end of holographic recording, i.e. the final condition of Eqn. 2.10d. In previous chapters the masses of binder b_0 and nanoparticles z_0 were expressed as relative to the initial mass of monomer m_0 within the domain of the grating, this convention will be extended to s_0. The domain $x \in [0, \Lambda]$ is periodic, i.e. $s(x + \Lambda, t) = s(x, t)$, etc. As in Chapter 3,

a periodic boundary condition can be modelled as

$$\frac{\partial^n s}{\partial x^n}(0,t) = \frac{\partial^n s}{\partial x^n}(\Lambda,t) \qquad n = \{0,1,2,...\},$$

$$\frac{\partial^n z^{(e)}}{\partial x^n}(0,t) = \frac{\partial^n z^{(e)}}{\partial x^n}(\Lambda,t) \qquad n = \{0,1,2,...\},$$

$$\frac{\partial^n z^{(s)}}{\partial x^n}(0,t) = \frac{\partial^n z^{(s)}}{\partial x^n}(\Lambda,t) \qquad n = \{0,1,2,...\}.$$

Eqn. 4.1 can be nondimensionalized by defining the following constants

$$x = \Lambda\bar{x}, \qquad t = t_0\bar{t}, \qquad s = s_0\bar{s}, \qquad z^{(e)} = z_0\overline{z^{(e)}}, \qquad z^{(s)} = z_0\overline{z^{(s)}},$$

$$\alpha_x = \frac{D_s t_0}{\Lambda^2}, \qquad \gamma_{s,s} = \gamma_s z_0 t_0, \qquad \omega_{s,s} = \omega_s t_0 \frac{z_0}{s_0}, \qquad \gamma_{s,z} = \gamma_s s_0 t_0, \qquad \omega_{s,z} = \omega_s t_0.$$

As with the numerical simulation for the formation of holographic gratings discussed in chapters 2 and 3 we can choose $t_0 = 1$ s. The characteristic times for analyte diffusion, capture and escape are defined such that their respective nondimensionalized parameter is equal to one

$$\tau_x = \frac{\Lambda^2}{D_s}, \qquad\qquad \tau_c = \frac{1}{\gamma_s z_0}, \qquad\qquad \tau_e = \frac{s_0}{z_0\omega_s}.$$

The results of a numerical simulation of Eqn. 4.1 are presented in Fig. 4.1. The theoretically modelled nanocomposite had a spatial frequency of 1000 lines/mm, a pre-exposure thickness of 49.1 μm and doped 5 wt.% with MFI nanozeolites.

(a) Vacant nanozeolites   (b) Occupied nanozeolites
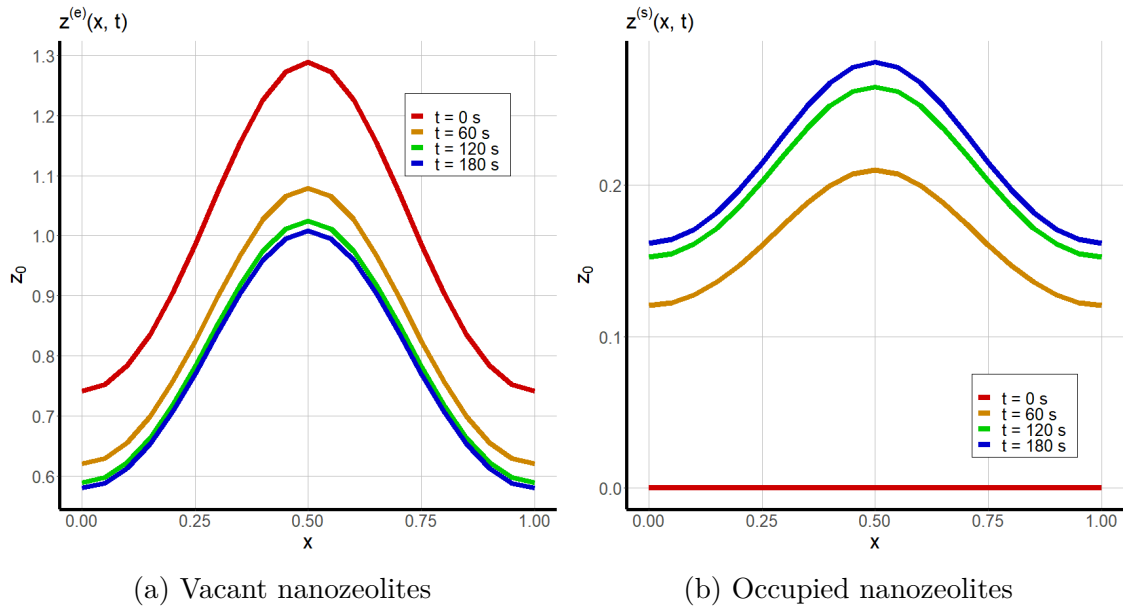
Fig. 4.1 Replacement of vacant nanozeolites with occupied nanozeolites.

At $t = 0$ seconds all nanozeolites are vacant, with continued exposure to the analyte the vacant nanozeolites are replaced with analyte-occupied nanozeolites. The replacement will continue until all available analyte has been captured. Note the second term in Eqn. 4.1a, because $z^{(e)} \to 0$ as $t \to \infty$ the rate of analyte capture falls off with time.

4.2   Change in Optical Properties

The change in the optical properties of a holographic grating due to the diffusion of analyte through the porous nanoparticles will be determined by several different things, namely, the quantity of analyte that passes into the structure; the RI of analyte n_s; the wt.% of nanoparticle doping; and the difference between the RI of nanozeolites filled with analyte molecules and the RI of vacant nanozeolites, $n_z^{(s)} - n_z^{(e)}$. The last of these is determined by the interior volume fraction and RI of the occupant. The RI of the nanocomposite as a function of x, y and t can be calculated via the same approach as in earlier chapters. The volume fractions are calculated by dividing solutions to Eqn. 4.1 by the appropriate density and from the final condition of numerical simulation of the formation of the holographic grating. The new RI spatial profile of the holographic grating $n(x, y, t)$ is calculated with the Lorentz-Lorenz equation,

$$\frac{n^2 - 1}{n^2 + 2} = \sum_j \phi_j \frac{n_j^2 - 1}{n_j^2 + 2}; \qquad j = \{b, m, p, q, z^{(e)}, z^{(s)}, s\},$$

whereby

$$\phi_i = \frac{V_i}{\sum_i V_i} \qquad i \in [b, m, p, q, z^{(e)}, z^{(s)}, s].$$

The new $n(x, t)$ is represented as a Fourier series, mean RI and RI modulation are modelled by the zero order and first order harmonic respectively (see Eqns. 1.6 and 1.5). Holographic gratings are recorded by the interference of an object and reference wave inside a photosensitive medium, there are two different geometries for this set-up: the object and reference wave can be incident on the same side of the photosensitive medium or they can be incident from opposite sides, the two geometries are called transmission and reflection respectively. Holographic images are reconstructed differently depending on the recording geometry, in viewing transmission holograms the grating is illuminated with monochromatic light, wavelength λ_r, and is diffracted at the Bragg angle,

$$\theta_B(t) = \sin^{-1}\left(\frac{\lambda_r}{2\overline{n}(t)\Lambda}\right) - \phi_r, \tag{4.2}$$

where $\overline{n}$ is the mean RI and ϕ_r is the slant angle. The viewing of reflection holograms is done by illuminating the grating with an optical beam at the Bragg

angle θ_B and the wavelength of the reconstructed light from the holographic grating will be determined by Bragg's law,

$$\lambda_r(t) = 2\overline{n}(t)\Lambda \sin\left(\theta_B(t) + \phi_r\right).$$

Alternatively, the reflection grating can be illuminated with white light and the color of reconstructed light will change according to the viewing angle. The other two geometries concerning holographic gratings are volume or planar (sometimes mislabelled as thick and thin). There are two established criteria to distinguish between the two. The first is the Klein-Cook criteria [15]

$$Q' = \frac{2\pi\lambda_r T}{\overline{n}\Lambda^2}. \tag{4.3}$$

For planar phase gratings described by Raman-Nath theory (thin), $Q' < 10$, whereas in volume gratings (thick) described by the Bragg theory $Q' > 10$; λ_r is the wavelength of the reconstruction beam (633 nm) and T is the thickness of the grating. The predicted classification of a theoretically modelled 10° slanted holographic grating recorded in a hybrid nanocomposite at an energy intensity of 80 mJ/cm^2 for spatial frequencies up to 1000 lines/mm and material thicknesses up to 100 μm are presented in Fig. 4.2.
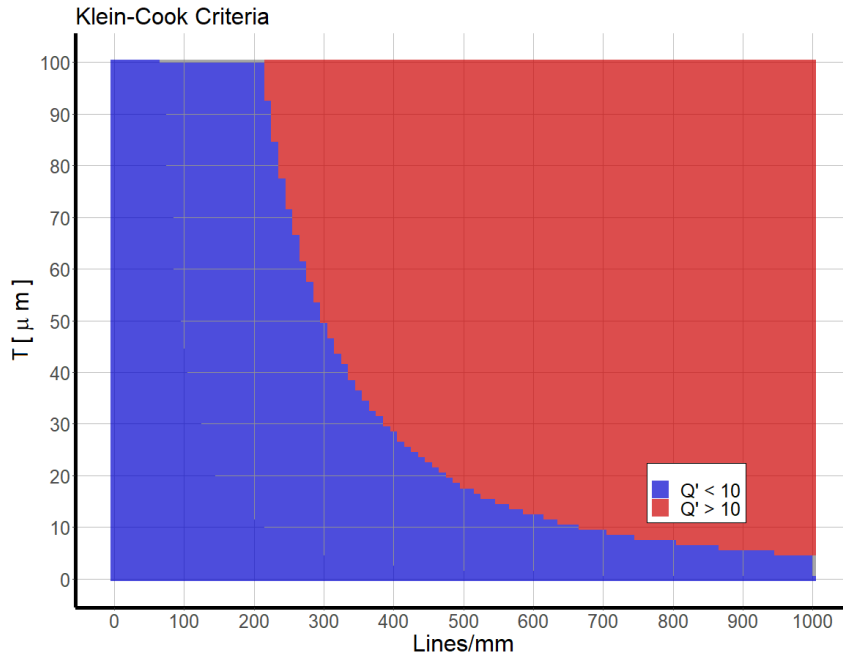
Fig. 4.2 Lines per mm on the x-axis, thickness in μm on y-axis

Fig. 4.2 shows that in holographic recording at spatial frequencies less than 200 lines/mm the predicted grating geometry will fall into the planar phase classification for any thickness. In this geometry the diffracted optical wavefront cannot redistribute its energy into the first diffraction order. Too much of the diffractive energy has leaked out into the higher order diffraction images. In this case the Bragg definition for

volume gratings is not satisfied and Eqn. 4.5 cannot model the diffraction efficiency. The optical properties of planar gratings will be discussed further in Section 4.4. The second established criteria for grating classification is the Moharam-Young [16]

$$\rho' = \frac{\lambda_r^2}{n\Delta n\Lambda^2}.$$
(4.4)

In planar phase gratings $\rho' \leq 1$, whereas holographic gratings fall into the volume phase classification if $\rho' > 1$. The predicted classification of unslanted holographic gratings using the Moharam-Young criteria is presented in Fig. 4.3.
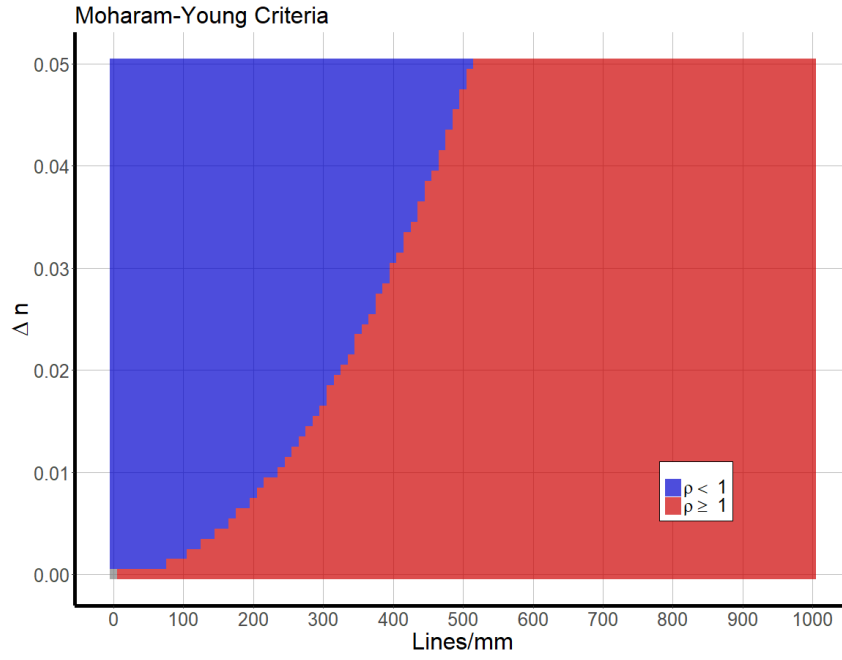
Fig. 4.3 Lines per mm on the x-axis, thickness in μm on y-axis

The predicted Moharam-Young value found a strong dependence on spatial frequency. This is not just due to the inverse square relationship between ρ' and Λ but also the fact that in numerical simulation of holographic recording, the first harmonic of RI is much smaller at reduced grating periods [2]. The diffraction efficiency of holographic gratings is very different for planar and volume phase geometries because of how the energy of diffracted light is distributed. For volume phase gratings most of the energy of the diffracted light is in the first diffraction order, the coupled-wave theory [82] gives the diffraction efficiency η of a volume phase transmission grating as

$$\eta(t) = \frac{\sin^2\sqrt{\xi^2 + \nu^2}}{\sqrt{1 + \xi^2/\nu^2}},$$
(4.5)

where

$$\xi(t) = \frac{\pi T}{\Lambda} \Delta\phi_r(t) \qquad \text{and} \qquad \nu(t) = \frac{\pi \Delta n(t) T}{\lambda_p \cos\theta_B(t)}.$$

The $\eta - \nu$ curve for an unslanted volume phase grating probed at the Bragg angle ($\xi = 0$) is shown in Fig. 4.4.
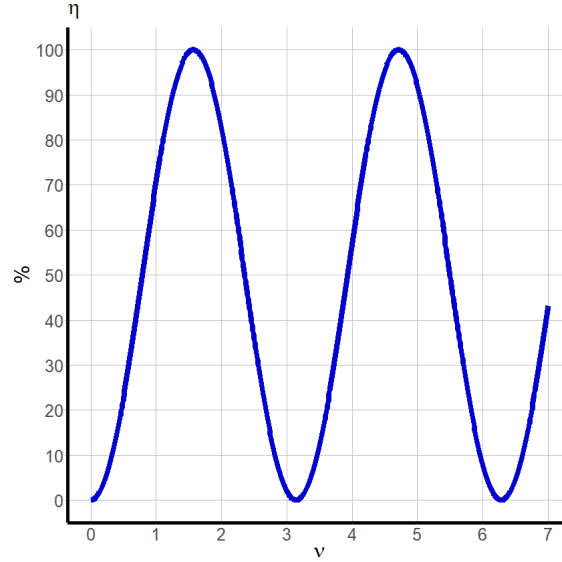
Fig. 4.4 Diffraction efficiency as a function of ν.

One can see that diffraction efficiency is most sensitive to changes in RI modulation and thickness at the inflection points of the curve,

$$\nu = \frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4}, \ldots$$

The predicted value for ν for spatial frequencies ranging from 100-1000 lines/mm, thickness 40-100 μm, nanoparticle doping 0-7.5 wt.% and recording intensities 2-20 mW/cm^2 for a fixed recording time of 100 seconds using the equations introduced in Chapter 2 are presented in Fig. 4.5

(a) 50 μm; 5 wt.%; 0.5 J/cm^2

(b) 475 lines/mm; 5 wt.%; 0.5 J/cm^2

(c) 475 lines/mm; 50 μm; 0.5 J/cm^2

(d) 475 lines/mm; 50 μm; 5 wt.%.
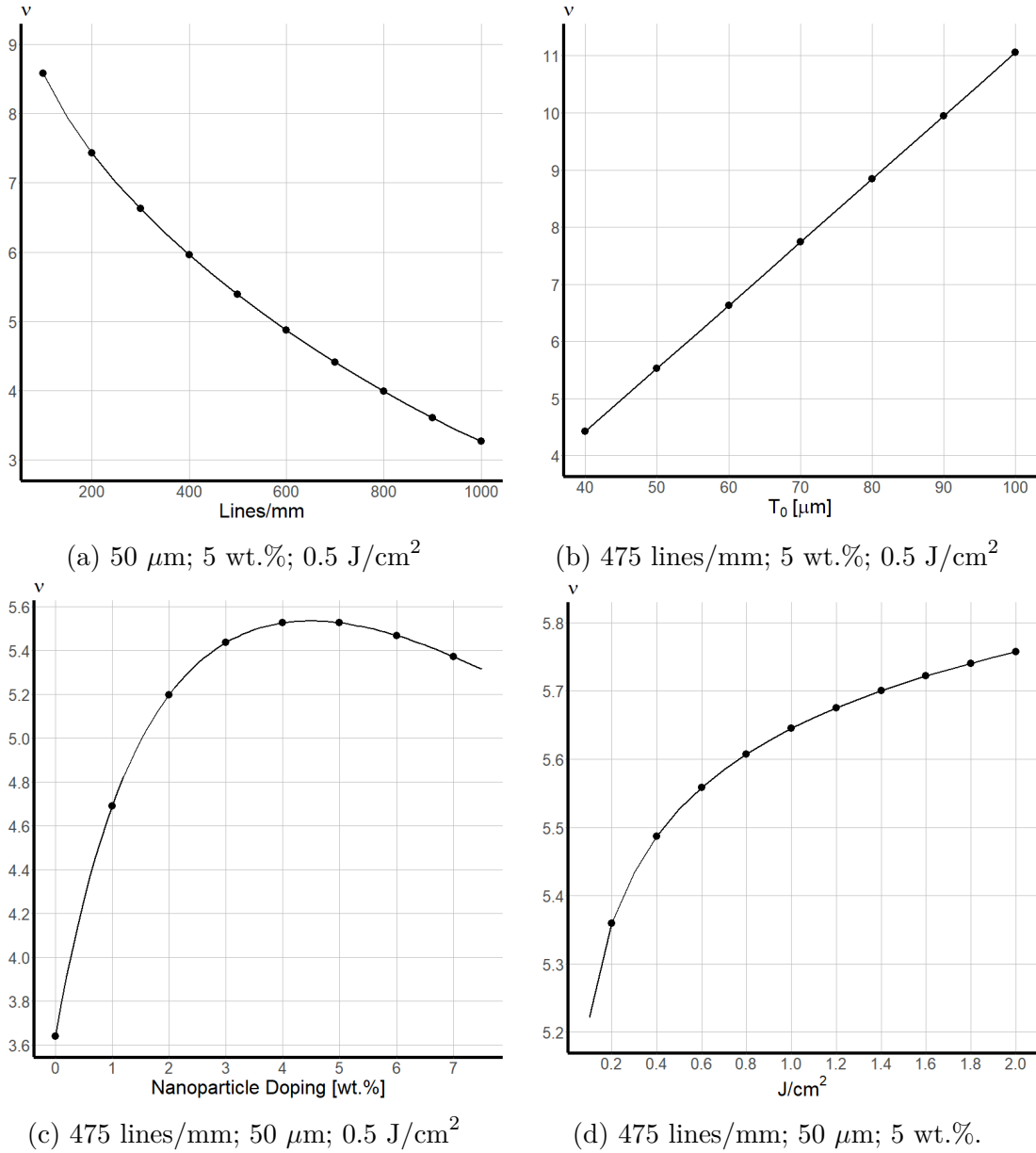
Fig. 4.5 Predicted ν for a variety of different initial conditions.

The results in Fig. 4.5 show that a theoretically modelled grating recorded in a AA/PVA photopolymer doped 5 wt.% with MFI nanozeolites ($n_z^{(e)} = 1.366$) with initial thickness 50 μm recorded with spatial frequency 475 lines/mm at an intensity 1 mW/cm^2 will have a predicted final RI modulation and thickness such that $\nu = 5.526 \approx 7\pi/4$. This theoretically modelled nanocomposite will be used as the initial condition for all results presented in this section. The predicted diffraction efficiency from numerical simulation of Eqn. 4.1 is plotted against time in Fig. 4.6 for a variety of different model parameters. The analyte to which the theoretically modelled nanocomposite is exposed is water, we chose this analyte because some of the relevant characteristics are known, namely $D_s = 2.3 \times 10^{-5}$ cm^2/s, $n_s = 1.33$ and $\rho_s = 1$ g/cm^3; there is some evidence [6] to suggest that $n_z^{(s)} = 1.46$.

(a) $n_z^{(s)} = 1.46$; $\tau_c = 60$ s; $\tau_e = 60$ s.

(b) $s_0/m_0 = 0.1$; $\tau_c = 60$ s; $\tau_e = 60$ s.

(c) $n_z^{(s)} = 1.46$; $s_0/m_0 = 0.1$; $\tau_e = 60$ s.

(d) $n_z^{(s)} = 1.46$; $s_0/m_0 = 0.1$; $\tau_c = 60$ s.
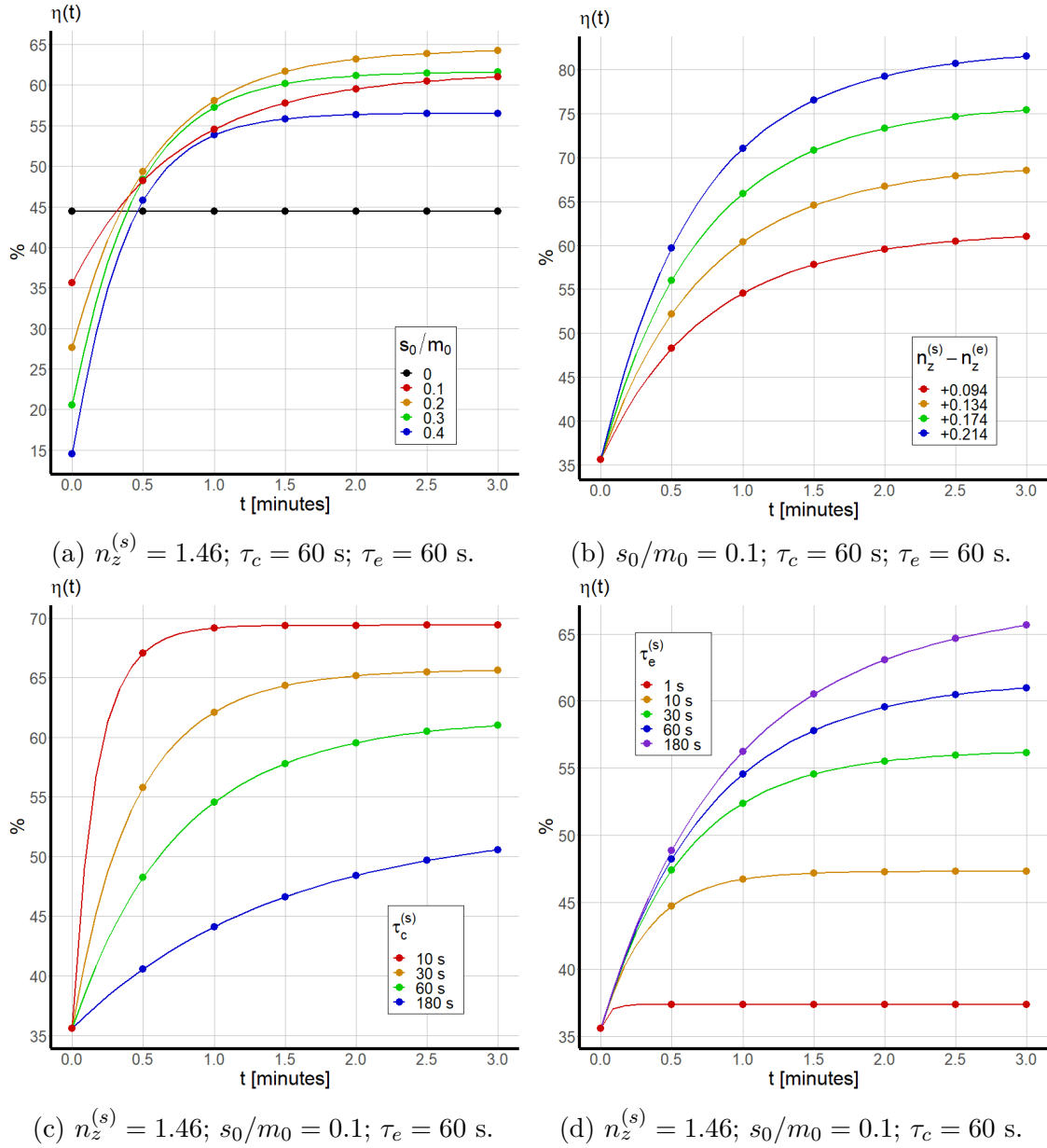
Fig. 4.6 Predicted diffraction efficiency of a theoretically modelled nanocomposite exposed to water.

One can see from Fig. 4.6a that the initial diffraction efficiency is strongly dependent on the quantity of analyte that enters the photonic structure. Recall that $t = 0$ is defined as the time at which diffusion of analyte has reached the far side of the structure and settled upon a new equilibrium. If there is a greater quantity of a low RI analyte inside the structure at $t = 0$ then the initial average RI of the structure, and hence initial diffraction efficiency, will be lower. If the values for $n_z^{(s)}$, τ_c and τ_e are known and established values then one can infer from the change in diffraction efficiency the quantity of analyte present in the photonic structure. Alternatively, if the quantity of analyte inside the photonic structure is known then one can use the change in diffraction efficiency to identify the analyte by its characteristic $n_z^{(s)}$ (see Fig. 4.6b). The times for capture and escape of analyte should be characteristic of both the analyte and the porous nanoparticle, one can see from Figs. 4.6c and 4.6d

that the time required for the system to settle upon a new equilibrium is strongly dependent on both τ_c and τ_e.

Nanozeolites are synthesized such that the shape and size of their pores can be fine tuned for the capture of specific molecules, ideally the nanozeolites would capture only the target analyte molecules but realistically some quantity of other analytes will also be captured. A photonic structure's affinity for the capture of a specific molecule over any other molecules it comes into contact with is referred to as its selectivity. In general, a holographic grating may be exposed to multiple different analytes. If the mass concentrations of P different analytes are labelled a_1, a_2, $\ldots a_P$ and nanozeolites occupied with the P different analytes are $z^{(a_1)}$, $z^{(a_2)}$, $\ldots z^{(a_P)}$ then the dynamics of the holographic sensor can be described by a system of $2P + 1$ partial differential equations,

$$\frac{\partial a_1}{\partial t} = D_{a_1} \frac{\partial^2 a_1}{\partial x^2} - \gamma_{a_1} a_1 z^{(e)} + \omega_{a_1} z^{(a_1)}, \tag{4.6a}$$

$$\frac{\partial a_2}{\partial t} = D_{a_2} \frac{\partial^2 a_2}{\partial x^2} - \gamma_{a_2} a_2 z^{(e)} + \omega_{a_2} z^{(a_2)}, \tag{4.6b}$$

$$\vdots$$

$$\frac{\partial a_P}{\partial t} = D_{a_P} \frac{\partial^2 a_P}{\partial x^2} - \gamma_{a_P} a_P z^{(e)} + \omega_{a_P} z^{(a_P)}, \tag{4.6c}$$

$$\frac{\partial z^{(e)}}{\partial t} = \sum_{i=1}^{P} -\gamma_{a_i} a_i z^{(e)} + \omega_{a_i} z^{(a_i)}, \tag{4.6d}$$

$$\frac{\partial z^{(a_1)}}{\partial t} = +\gamma_{a_1} a_1 z^{(e)} - \omega_{a_1} z^{(a_1)}, \tag{4.6e}$$

$$\frac{\partial z^{(a_2)}}{\partial t} = +\gamma_{a_2} a_2 z^{(e)} - \omega_{a_2} z^{(a_2)}, \tag{4.6f}$$

$$\vdots$$

$$\frac{\partial z^{(a_P)}}{\partial t} = +\gamma_{a_P} a_P z^{(e)} - \omega_{a_P} z^{(a_P)}. \tag{4.6g}$$

We will restrict our analysis in this chapter to just two analytes, a target analyte a and a secondary analyte s. In the mathematical model described here selectivity can be measured by the ratio of γ_a to γ_s and ω_a to ω_s.

$$\frac{\partial a}{\partial t} = D_a \frac{\partial^2 a}{\partial x^2} - \gamma_a a z^{(e)} + \omega_a z^{(a)}, \tag{4.7a}$$

$$\frac{\partial s}{\partial t} = D_s \frac{\partial^2 s}{\partial x^2} - \gamma_s s z^{(e)} + \omega_s z^{(s)}, \tag{4.7b}$$

$$\frac{\partial z^{(e)}}{\partial t} = -\gamma_a a z^{(e)} + \omega_a z^{(a)} - \gamma_s s z^{(e)} + \omega_s z^{(s)}, \tag{4.7c}$$

$$\frac{\partial z^{(a)}}{\partial t} = +\gamma_a a z^{(e)} - \omega_a z^{(a)}, \tag{4.7d}$$

$$\frac{\partial z^{(s)}}{\partial t} = +\gamma_s s z^{(e)} - \omega_s z^{(s)}. \tag{4.7e}$$

The domain is the same as Eqn. 4.1, the additional initial and boundary conditions required are

$$a(x, 0) = a_0, \qquad\qquad z^{(a)}(x, 0) = 0,$$

$$\frac{\partial^n a}{\partial x^n}(0, t) = \frac{\partial^n a}{\partial x^n}(\Lambda, t) \qquad\qquad n = \{0, 1, 2, ...\},$$

$$\frac{\partial^n z^{(a)}}{\partial x^n}(0, t) = \frac{\partial^n z^{(a)}}{\partial x^n}(\Lambda, t) \qquad\qquad n = \{0, 1, 2, ...\}.$$

The additional characteristic times for analyte diffusion, capture and escape are

$$\tau_x^{(s)} = \frac{\Lambda^2}{D_s}, \qquad\qquad \tau_c^{(s)} = \frac{1}{\gamma_s z_0}, \qquad\qquad \tau_e^{(s)} = \frac{s_0}{z_0 \omega_s},$$

$$\tau_x^{(a)} = \frac{\Lambda^2}{D_a}, \qquad\qquad \tau_c^{(a)} = \frac{1}{\gamma_a z_0}, \qquad\qquad \tau_e^{(a)} = \frac{a_0}{z_0 \omega_a}.$$

The diffraction efficiency of a theoretically modelled holographic grating ($\nu = 5.526$) exposed to two analytes is plotted against time in Fig. 4.7 for $a_0/s_0 \in [0, 0.4]$ and for both weak ($\tau_c^{(a)} = \tau_c^{(s)}$) and strong ($\tau_c^{(a)} = 0.1\tau_c^{(s)}$) selectivity. As there are already several unknown variables we will simplify the analysis by assuming that captured analyte molecules will remain trapped indefinitely, i.e. $\tau_e^{(s)}$, $\tau_e^{(a)} = \infty$.

(a) Weak capture selectivity $\qquad$ (b) Strong capture selectivity
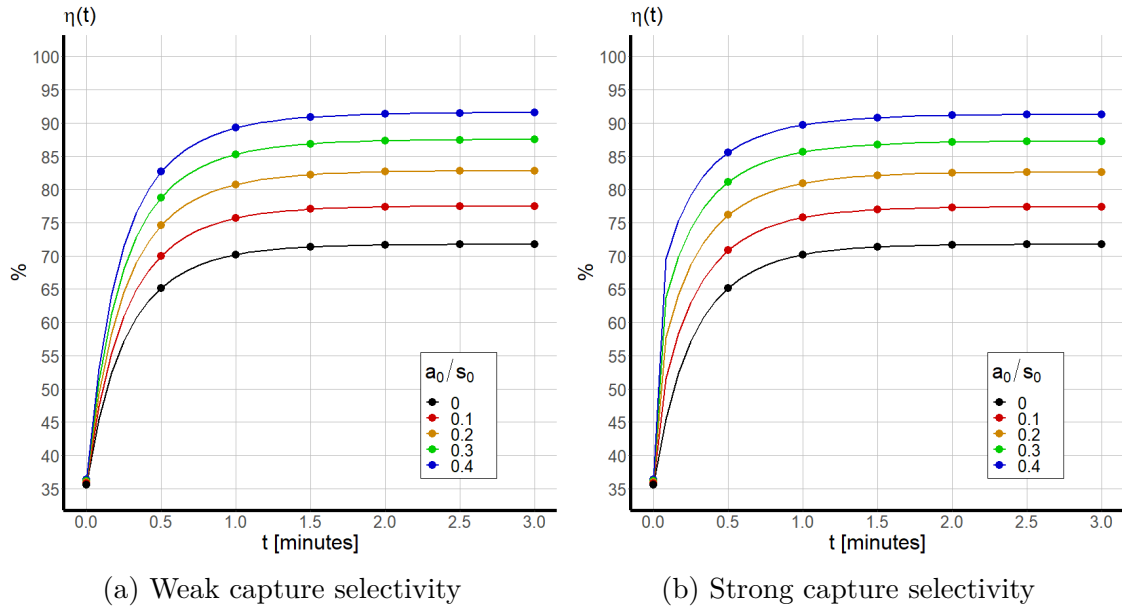
Fig. 4.7 Diffraction efficiency response to concentration of a second analyte; $n_a = 2.1$; $n_z^{(a)} = 1.6$; $D_a = 1 \times 10^{-7}$ cm^2/s; $\rho_a = 9$ g/cm^3.

The results presented in Fig. 4.7 show that although capture selectivity has some effect on diffraction efficiency in the early stage of the response, it makes no difference to the long term response, i.e. after all vacant nanozeolites have been replaced with occupied nanozeolites. This is because only a finite quantity of both analytes enter the nanocomposite and a_0, $s_0 < z_0$. After a sufficient length of time has passed, all analytes will be captured so selectivity does not influence the final response. However, if the holographic grating is exposed, not instantaneously, but continuously to a reservoir of fluid consisting of the two analytes one might expect a very different response. To model this scenario it will be necessary to extend our analysis to a second spatial dimension.

4.3 Diffusion of Multiple Analytes within a Holographic Grating with Finite Thickness

As in Chapter 3 the y-axis is defined along the depth of the nanocomposite. Extending Eqn. 4.7 to a second spatial dimension, the governing equations modelling the change in optical properties of a holographic grating in response to exposure to multiple analytes will be

$$\frac{\partial a}{\partial t} = D_a \frac{\partial^2 a}{\partial x^2} + D_a \frac{\partial^2 a}{\partial y^2} - \gamma_a a z^{(e)} + \omega_a z^{(a)}, \tag{4.8a}$$

$$\frac{\partial s}{\partial t} = D_s \frac{\partial^2 s}{\partial x^2} + D_s \frac{\partial^2 s}{\partial y^2} - \gamma_s s z^{(e)} + \omega_s z^{(s)}, \tag{4.8b}$$

$$\frac{\partial z^{(e)}}{\partial t} = -\gamma_a a z^{(e)} + \omega_a z^{(a)} - \gamma_s s z^{(e)} + \omega_s z^{(s)}, \tag{4.8c}$$

$$\frac{\partial z^{(a)}}{\partial t} = +\gamma_a a z^{(e)} - \omega_a z^{(a)}, \tag{4.8d}$$

$$\frac{\partial z^{(s)}}{\partial t} = +\gamma_s s z^{(e)} - \omega_s z^{(s)}. \tag{4.8e}$$

The spatial domain for analysis of the holographic sensor is the same domain used in the formation of the grating described in Chapter 3

$$0 \le x \le \hat{x}, \qquad\qquad 0 \le y \le T_1, \qquad\qquad t \ge 0.$$

Here, T_1 is the thickness of the nanocomposite after holographic recording, the subscript 1 is used to distinguish it from T_0 the pre-recording thickness. As we restrict our analysis to unslanted holographic gratings, $\hat{x} = \Lambda_1$. The domain $y \in [0, T_1]$ is fixed for all $t \ge 0$. Realistically, the influx of analyte material will result in swelling in the thickness of the holographic layer, we assume that the change in physical dimensions is sufficiently small that its influence on diffraction efficiency is negligible; this will be discussed further in Chapter 5. At $y = 0$ the holographic grating is fixed to an immovable glass backing so the zero-flux boundary condition holds

$$\frac{\partial a}{\partial y}(x, 0, t) = \frac{\partial s}{\partial y}(x, 0, t) = \frac{\partial z^{(e)}}{\partial y}(x, 0, t) = \frac{\partial z^{(a)}}{\partial y}(x, 0, t) = \frac{\partial z^{(s)}}{\partial y}(x, 0, t) = 0.$$

At $y = T_1$ there is no movement of nanozeolites across the boundary so the zero-flux boundary condition also holds here

$$\frac{\partial z^{(e)}}{\partial y}(x, T_1, t) = \frac{\partial z^{(a)}}{\partial y}(x, T_1, t) = \frac{\partial z^{(s)}}{\partial y}(x, T_1, t) = 0.$$

There is transfer of analyte across the boundary from outside the domain to within, the material flux can be modelled by a radiation boundary condition [83]

$$-D_a \frac{\partial a}{\partial y}(x, T_1, t) = k \left[a(x, T_1, t) - a_0 \right], \tag{4.9a}$$

$$-D_s \frac{\partial s}{\partial y}(x, T_1, t) = k \left[s(x, T_1, t) - s_0 \right], \tag{4.9b}$$

where a_0 and s_0 are the respective analyte concentrations in the fluid to which
the holographic grating is exposed and k is a constant governing the permeability of
the boundary. The holographic grating is exposed to a volume of analyte so much
larger than the volume of the grating itself that it can be modelled as an infinite
reservoir, hence the mass of target analyte at the surface will be a constant, a_0 and
s_0. Earlier theoretical studies in modelling diffusion of aqueous solutions within a
nanocomposite [84, 85] use the following initial and boundary conditions

$$a(x, y, 0) = 0, \qquad\qquad 0 \le y < T_1, \qquad (4.10\text{a})$$

$$a(x, T_1, 0) = a_0, \qquad\qquad\qquad (4.10\text{b})$$

$$s(x, y, 0) = 0, \qquad\qquad 0 \le y < T_1, \qquad (4.10\text{c})$$

$$s(x, T_1, 0) = s_0, \qquad\qquad\qquad (4.10\text{d})$$

$$z^{(e)}(x, y, 0) = z(x, y), \qquad\qquad\qquad (4.10\text{e})$$

$$z^{(a)}(x, y, 0) = 0, \qquad\qquad\qquad (4.10\text{f})$$

$$z^{(s)}(x, y, 0) = 0, \qquad\qquad\qquad (4.10\text{g})$$

$$a(x, T_1, t) = a_0, \qquad\qquad t \ge 0, \qquad (4.10\text{h})$$

$$s(x, T_1, t) = s_0, \qquad\qquad t \ge 0, \qquad (4.10\text{i})$$

where $z(x, y)$ is the final condition from numerical simulation of Eqn. 3.3d. The
three varieties of the nanozeolites in the holographic sensor model form a conserved
system,

$$z^{(e)}(x, y, t) + z^{(a)}(x, y, t) + z^{(s)}(x, y, t) = z(x, y).$$

A consequence of Eqns. 4.10h and 4.10i is that Eqn. 4.9 is reduced to a simple
zero-flux boundary condition. Eqn. 4.8 can be nondimensionalized by defining the
following constants

$$x = \hat{x}\overline{x}, \qquad\qquad y = T_1\overline{y}, \qquad\qquad t = t_0\overline{t}$$

$$a = a_0\overline{a}, \qquad s = s_0\overline{s}, \qquad z^{(e)} = z_0\overline{z^{(e)}}, \qquad z^{(a)} = z_0\overline{z^{(a)}}, \qquad z^{(s)} = z_0\overline{z^{(s)}},$$

$$\alpha_{s,x} = \frac{D_s t_0}{\hat{x}^2}, \qquad \alpha_{a,x} = \frac{D_a t_0}{\hat{x}^2}, \qquad \alpha_{s,y} = \frac{D_s t_0}{T_1^2}, \qquad \alpha_{a,y} = \frac{D_a t_0}{T_1^2},$$

$$\gamma_{a,a} = \gamma_a z_0 t_0 \qquad \omega_{a,a} = \omega_a t_0 \frac{z_0}{a_0}, \qquad \gamma_{s,s} = \gamma_s z_0 t_0, \qquad \omega_{s,s} = \omega_s t_0 \frac{z_0}{s_0},$$

$$\gamma_{a,z} = \gamma_a a_0 t_0, \qquad \omega_{a,z} = \omega_a t_0, \qquad \gamma_{s,z} = \gamma_s s_0 t_0, \qquad \omega_{s,z} = \omega_s t_0.$$

The nondimensionalized governing equations become

$$\frac{\partial a}{\partial t} = \alpha_{a,x} \frac{\partial^2 a}{\partial x^2} + \alpha_{a,y} \frac{\partial^2 a}{\partial y^2} - \gamma_{a,a} a z^{(e)} + \omega_{a,a} z^{(a)}, \tag{4.11a}$$

$$\frac{\partial s}{\partial t} = \alpha_{s,x} \frac{\partial^2 s}{\partial x^2} + \alpha_{s,y} \frac{\partial^2 s}{\partial y^2} - \gamma_{s,s} s z^{(e)} + \omega_{s,s} z^{(s)}, \tag{4.11b}$$

$$\frac{\partial z^{(e)}}{\partial t} = -\gamma_{a,z} a z^{(e)} + \omega_{a,z} z^{(a)} - \gamma_{s,z} s z^{(e)} + \omega_{s,z} z^{(s)}, \tag{4.11c}$$

$$\frac{\partial z^{(a)}}{\partial t} = \gamma_{a,z} a z^{(e)} - \omega_{a,z} z^{(a)}, \tag{4.11d}$$

$$\frac{\partial z^{(s)}}{\partial t} = \gamma_{s,z} s z^{(e)} - \omega_{s,z} z^{(s)}. \tag{4.11e}$$

and the domain is now

$$0 \le x \le 1, \qquad\qquad 0 \le y \le 1, \qquad\qquad t \ge 0.$$

The boundary conditions are as follows,

$$\frac{\partial^n a}{\partial x^n}(0, y, t) = \frac{\partial^n a}{\partial x^n}(1, y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$
$$\frac{\partial a}{\partial y}(x, 0, t) = \frac{\partial a}{\partial y}(x, 1, t) = 0,$$
$$a(x, 1, t) = 1,$$

$$\frac{\partial^n s}{\partial x^n}(0, y, t) = \frac{\partial^n s}{\partial x^n}(1, y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$
$$\frac{\partial s}{\partial y}(x, 0, t) = \frac{\partial s}{\partial y}(x, 1, t) = 0,$$
$$s(x, 1, t) = 1,$$

$$\frac{\partial^n z^{(e)}}{\partial x^n}(0, y, t) = \frac{\partial^n z^{(e)}}{\partial x^n}(1, y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$
$$\frac{\partial z^{(e)}}{\partial y}(x, 0, t) = \frac{\partial z^{(e)}}{\partial y}(x, 1, t) = 0,$$

$$\frac{\partial^n z^{(a)}}{\partial x^n}(0, y, t) = \frac{\partial^n z^{(a)}}{\partial x^n}(1, y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$

$$\frac{\partial z^{(a)}}{\partial y}(x, 0, t) = \frac{\partial z^{(a)}}{\partial y}(x, 1, t) = 0,$$

$$\frac{\partial^n z^{(s)}}{\partial x^n}(0, y, t) = \frac{\partial^n z^{(s)}}{\partial x^n}(1, y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$

$$\frac{\partial z^{(s)}}{\partial y}(x, 0, t) = \frac{\partial z^{(s)}}{\partial y}(x, 1, t) = 0,$$

The initial conditions are

$$a(x, 1, 0) = 1,$$
$$a(x, y, 0) = 0 \qquad\qquad\qquad 0 \le y < 1,$$
$$s(x, 1, 0) = 1,$$
$$s(x, y, 0) = 0 \qquad\qquad\qquad 0 \le y < 1,$$
$$z^{(e)}(x, y, 0) = z(x, y),$$
$$z^{(a)}(x, y, 0) = 0,$$
$$z^{(s)}(x, y, 0) = 0.$$

There is not yet an established empirical method for estimating s_0 or a_0 but one can compare the results for numerical simulation of the model with experimental observations. From the definition of the nondimensionalized constants one can define the characteristic time for analyte diffusion in the y-direction

$$\tau_y^{(s)} = \frac{T_1^2}{D_s}, \qquad\qquad\qquad \tau_y^{(a)} = \frac{T_1^2}{D_a}.$$

The change in optical properties during exposure to analyte is dependent on s_0, the difference in RI of vacant and occupied nanozeolites and the characteristic times for the diffusion, capture and escape of analyte. Assuming that $s_0/m_0 = 0.2$, the predicted change in RI modulation of a theoretically modelled nanocomposite ($\nu = 5.56$) exposed to deionized water is plotted against exposure time in Fig. 4.8. Numerical simulations are run for scenarios in which the characteristic time for escape is less than, equal to and greater than the characteristic time for capture and for scenarios in which the characteristic time for capture is less than, equal to and greater than the characteristic time for diffusion, i.e. $0 \le \tau_c^{(s)}, \tau_e^{(s)} \le 10\tau_y^{(s)}$.
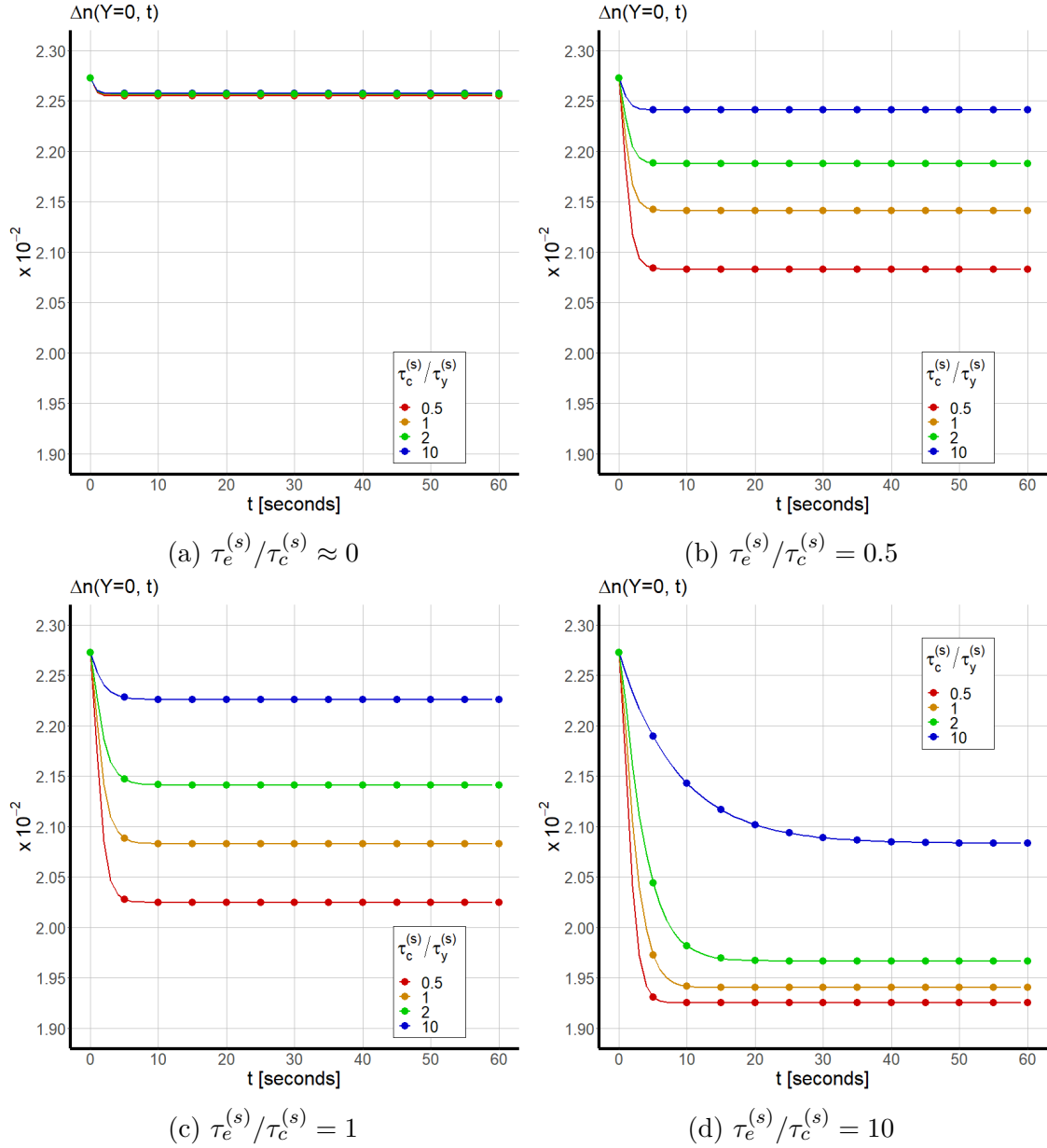
Fig. 4.8 Change in RI modulation of volume phase grating recorded in a hybrid photopolymer material exposed to deionized water.

In the results presented above, the analyte (deionized water) has a uniform grating profile in the x-direction so the effect of its addition to the nanocomposite is to lower the binder RI. RI modulation is due to the non-uniform grating profile of cross-linked polymer chains and inorganic nanoparticles. This part of the nanocomposite accounts for a smaller fraction of the volume as more analyte enters the nanocomposite and moves through the depth of the structure, hence there is a drop in RI modulation which continues until the mass concentration of analyte at the far side ($y = 0$) reaches equilibrium with the exposed surface ($y = T_1$). For deionized water $D_s = 2.3 \times 10^{-5}$ cm^2/s, thus in a 50 μm thick nanocomposite $\tau_y^{(s)} = 1.09$ seconds. Fig. 4.8 shows that for $\tau_e^{(s)} = 0$ seconds, which is to say the analyte molecules escape instantly, there is no significant change in Δn regardless of how little time is required for the capture of analyte molecules. As the ratio of $\tau_c^{(s)}$ to $\tau_y^{(s)}$ increases, the nanocomposite

takes longer to settle on a new equilibrium. The ratio $\tau_e^{(s)}$ to $\tau_c^{(s)}$, a measure of the nanozeolites affinity to retain an analyte molecule, is an important factor in determining the change in diffraction efficiency for a given quantity of analyte s_0 and a given difference between $n_z^{(s)}$ and $n_z^{(e)}$. An increase in the total mass of analyte s_0 that has permeated into the holographic grating means a change in the nondimensionalized parameters $\gamma_{s,z}$ (increasing) and $\omega_{s,s}$ (decreasing) which in turn increases the ratio of analyte-occupied to vacant nanozeolites. Increasing the RI of the analyte means an increase in the RI of analyte-occupied nanozeolites accumulated at the dark fringes which reduces Δn. The heatmap of the predicted change in Δn before and after a 60 second exposure to a deionized water in Fig. 4.9 offers a more comprehensive illustration of how the characteristic times for the capture and escape of analyte influence the response.
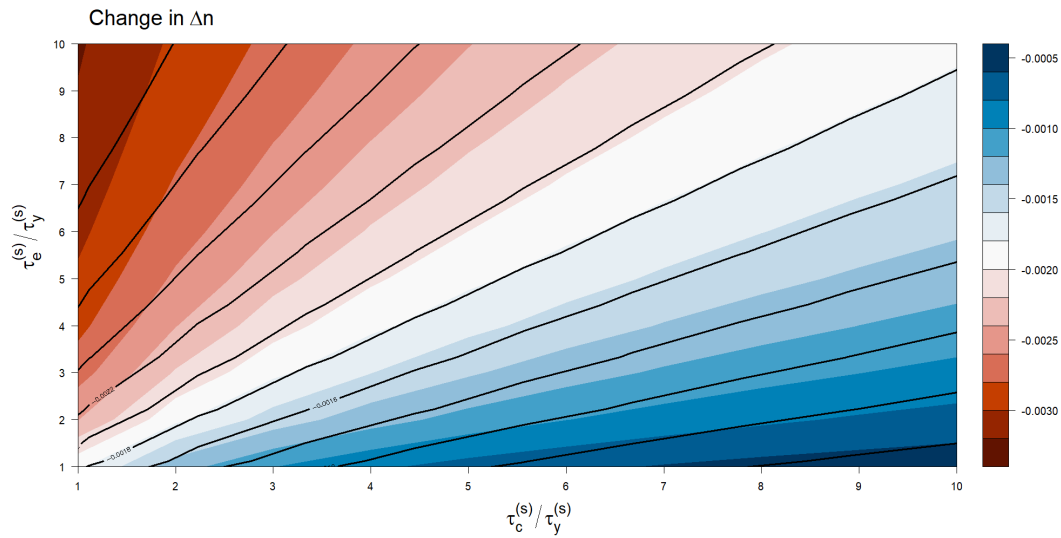
Fig. 4.9 Predicted change in RI modulation of a holographic grating recorded on a hybrid photopolymer after 60 s exposure to deionized water.

The theoretical model can also simulate the change in diffraction efficiency for different scenarios in which the holographic grating is exposed to multiple analytes. The first set of scenarios relate to the capture of the target analyte, a feature of a good environmental sensor is that the characteristic time for the capture of a target analyte should be less than the time for the capture of other analytes, $\tau_c^{(a)} << \tau_c^{(s)}$; this condition simply requires $\gamma_a >> \gamma_s$. Another set of scenarios relate to the characteristic time for the escape of the target analyte relative to the time for the escape of secondary analytes. The predicted change in RI modulation of a holographic grating ($\nu = 5.56$) exposed for 60 seconds to an unknown analyte ($\rho_a = 9$ g/cm^3, $n_a = 2$, $n_z^{(a)} = 1.6$, $D_a = D_s$, $a_0 = s_0$) and deionized water is plotted against time in Figs. 4.10 where the ratio of escape time is 1, 2 and 10.

(a) Weak selectivity, $\tau_c^{(a)} = \tau_c^{(s)}$ (b) Strong selectivity, $\tau_c^{(a)} = 0.1\tau_c^{(s)}$
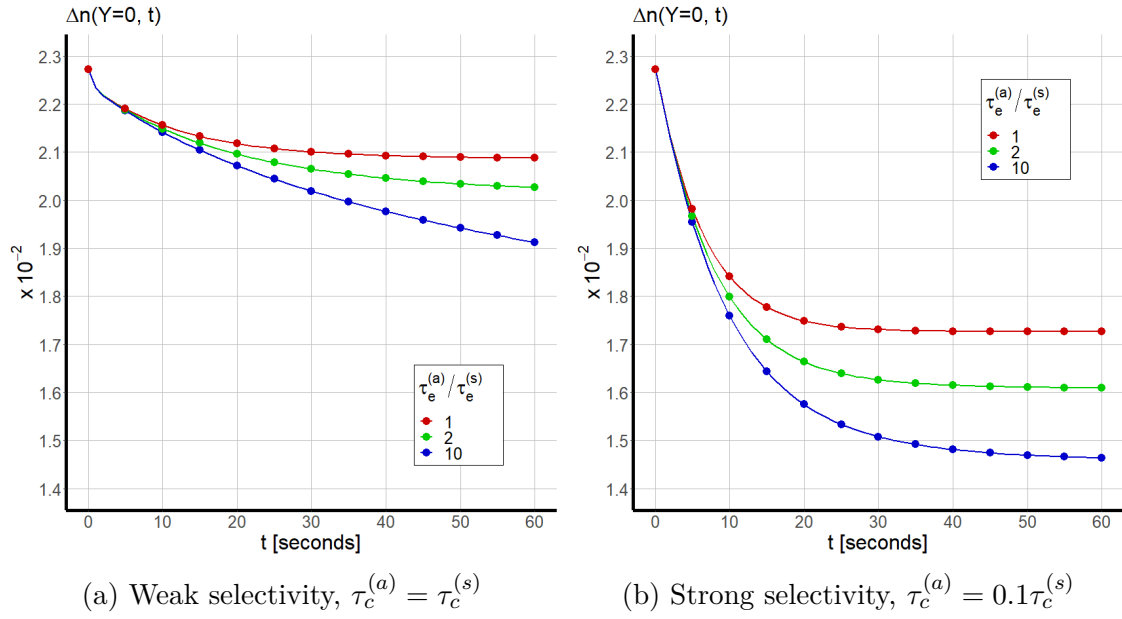
Fig. 4.10 Change in RI modulation of a holographic grating ($\nu = 5.56$) exposed to a target analyte and deionized water.

A photonic structure which captures target analyte molecules with equal probability, i.e. $\tau_c^{(s)} \approx \tau_c^{(a)}$, is said to have poor selectivity; a holographic grating that captures almost exclusively the target analyte $\tau_c^{(a)} << \tau_c^{(s)}$ demonstrates good selectivity and is ideal for environmental sensing applications. Figs. 4.10a and 4.10b show that the change in optical properties of a holographic sensor exposed to a target analyte is a function of not just the affinity of the nanozeolites to capture analyte molecules (selectivity) but also on their ability to keep the target analyte molecule trapped whilst allowing the escape of the secondary analyte molecule. In simulations of fast analyte escape ($\tau_e^{(a)} = \tau_e^{(s)}$) there is very little change in optical properties in response to exposure, even in simulation of strong selectivity. Ideally, a holographic sensor would capture only target analyte and trap the analyte molecules indefinitely; this ideal case can be modelled by $\tau_c^{(s)} >> \tau_c^{(a)}$ and $\tau_e^{(s)} << \tau_e^{(a)}$. If the RI of analyte-occupied nanozeolites together with the characteristic times for the capture and escape of a target analyte are all known values, a_0 can be quantified via a comparison of the predicted change in optical properties of an ideal holographic sensor exposed to a target analyte with experimental results. The predicted change in RI modulation and diffraction efficiency for $0 \leq a_0/s_0 \leq 0.6$ is presented in Fig. 4.11a. Alternatively, if the saturated mass concentration of analyte a_0 is known then the RI of analyte-occupied nanozeolites can be deduced from the change in optical properties (see Fig. 4.11b).

(a) $n_z^{(a)} - n_z^{(s)} = 0.14$ (b) $a_0/s_0 = 0.1$
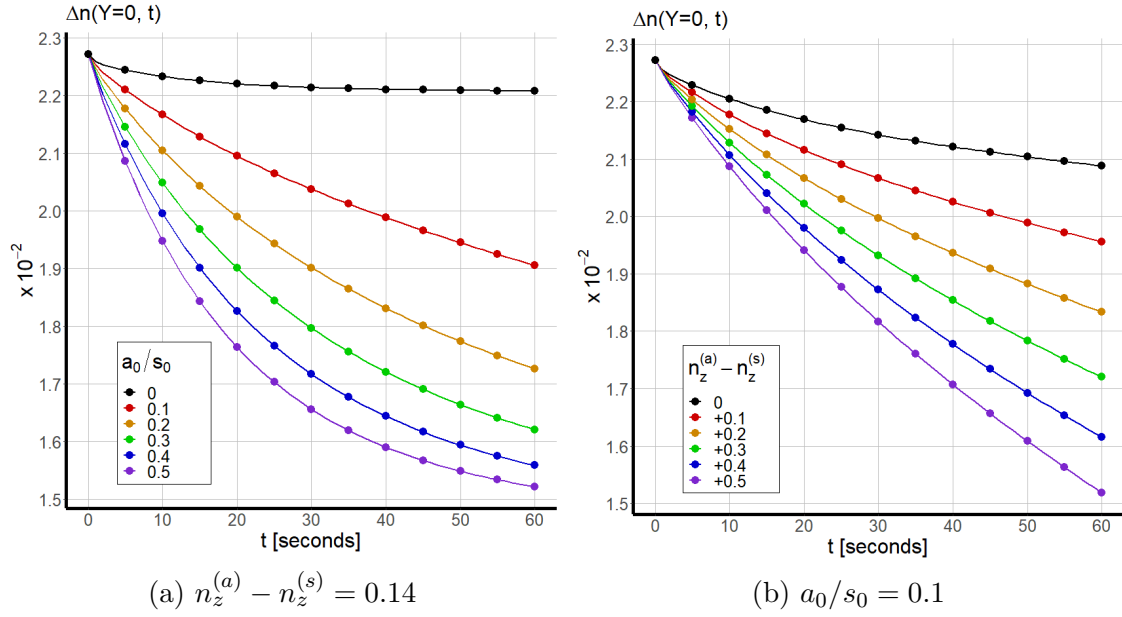
Fig. 4.11 Change in RI modulation of an ideal holographic sensor ($\nu = 5.56$) exposed to a target analyte and deionized water ($\tau_y^{(s)} = \tau_y^{(a)} = 1$ s; $\tau_c^{(s)} = 60$ s; $\tau_e^{(s)} = 10$ s; $\tau_c^{(a)} = 6$ s; $\tau_e^{(a)} = 100$ s).

By increasing the concentration of analyte, and hence a_0, there is a larger fraction of vacant nanozeolites which can become occupied producing a larger change in Δn for a given exposure duration. The system reaches saturation once all vacant nanozeolites have been replaced with the higher RI occupied nanozeolites. By increasing a_0 the concentration gradient and hence diffusion of analyte is increased which reduces the time required for saturation. The difference between the RI of vacant and occupied nanozeolites is characteristic of the target analyte itself. A greater difference between $n_z^{(e)}$ and $n_z^{(a)}$ is predicted to yield a greater change in diffraction efficiency for a given exposure duration. Results from numerical simulations of the model with different ρ_a and ρ_s found no noticeable influence on the predicted change in diffraction efficiency.

4.4   Planar Phase Gratings

Due to their higher sensitivity and theoretically high diffraction efficiency, most of the earlier work on holographic sensors has been focused on volume holograms, but interest in planar phase holographic gratings, or more specifically Surface Relief Gratings (SRG), is growing [86, 5, 1] mainly due to their fast response time. The distinction between SRGs and standard planar phase gratings is in the post-recording treatment. After the unslanted holographic grating is recorded in a photopolymer media the photonic structure is placed in an oven for thermal treatment. The effect of thermal treatment is to remove any bulk material, leaving behind only the cross-linked polymer, the remaining structure is just a few hundred nanometers thick. Next, the polymer structure is spin coated with nanozeolites incorporated in a sol gel, after the spin coating is complete the troughs between the polymer will be filled in by

the nanozeolites. The sensor output is the result of interaction between analyte and surface structure so target analytes do not need to permeate into the structure, this gives SRGs the potential for faster response times and improved selectivity. Although the equations introduced in Chapter 3 cannot model the thermal treatment nor the spin coating process it is still possible to model the formation of a planar phase grating with similar optical properties and physical dimensions. Numerical simulation of the equations described in this chapter can be applied to model the response of a theoretically modelled SRG exposed to a target analyte. If the Klein-Cook (Eqn. 4.3) value is less than 10, or the Moharam-Young (Eqn. 4.4) value is less than 1, the holographic grating is said to have a planar geometry and its optical properties are described by Raman-Nath theory [12]. In planar phase gratings the energy of the diffracted light is more dispersed amongst higher diffraction orders, Raman-Nath theory gives the diffraction efficiency of the m diffraction order from planar phase gratings as

$$\eta_m(y,t) = J_m^2 \left[\frac{\pi \Delta n(y,t) T_1}{\lambda_r \cos \theta_B(t)} \right], \tag{4.12}$$

where J_m is the Bessel function of order m which can be expressed as

$$J_m(x) = \sum_{p=0}^{\infty} \frac{(-1)^p}{p!(p+m)!} \left(\frac{x}{2} \right)^{2p+m} .$$

The $\eta_1 - \nu$ curve for an unslanted planar phase grating probed at the Bragg angle ($\xi = 0$) is shown in Fig. 4.12.
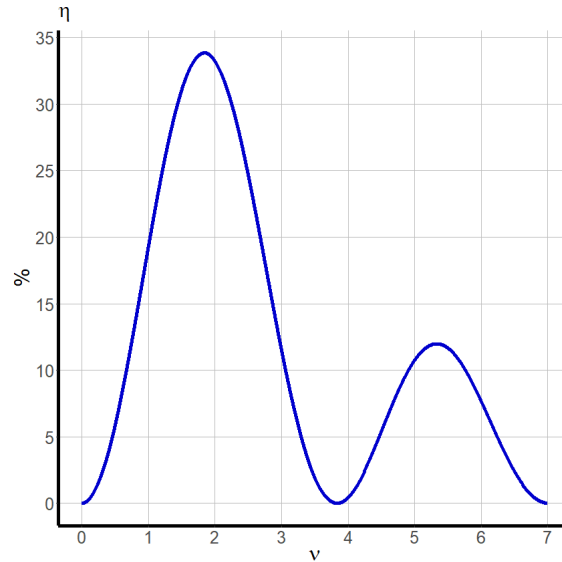
Fig. 4.12 Diffraction efficiency of the first diffraction order as a function of ν.

One can see that a planar phase grating is most sensitive for $\nu < 3.8$. The results in Fig. 4.2 show the conditions that will result in a predicted planar geometry, working within these parameters one can run numerical simulations of the equations from Chapter 3 with the view to predicting the recording conditions and sample preparation

required for the formation of a planar phase grating with optical properties and physical dimensions such that $\nu < 3.8$. The predicted ν at the end of a 500 mJ/cm^2 recording for initial thickness 5-40 μm, spatial frequencies 50-200 lines/mm, MFI nanozeolite doping 0-5 wt.% and recording intensities 1-15 mW/cm^2 is presented in Fig. 4.13.

(a) 100 lines/mm; 5 wt.%; 5 mW/cm^2

(b) 10 μm; 5 wt.%; 5 mW/cm^2

(c) 10 μm; 100 lines/mm; 5 mW/cm^2

(d) 10 μm; 100 lines/mm; 5 wt.%
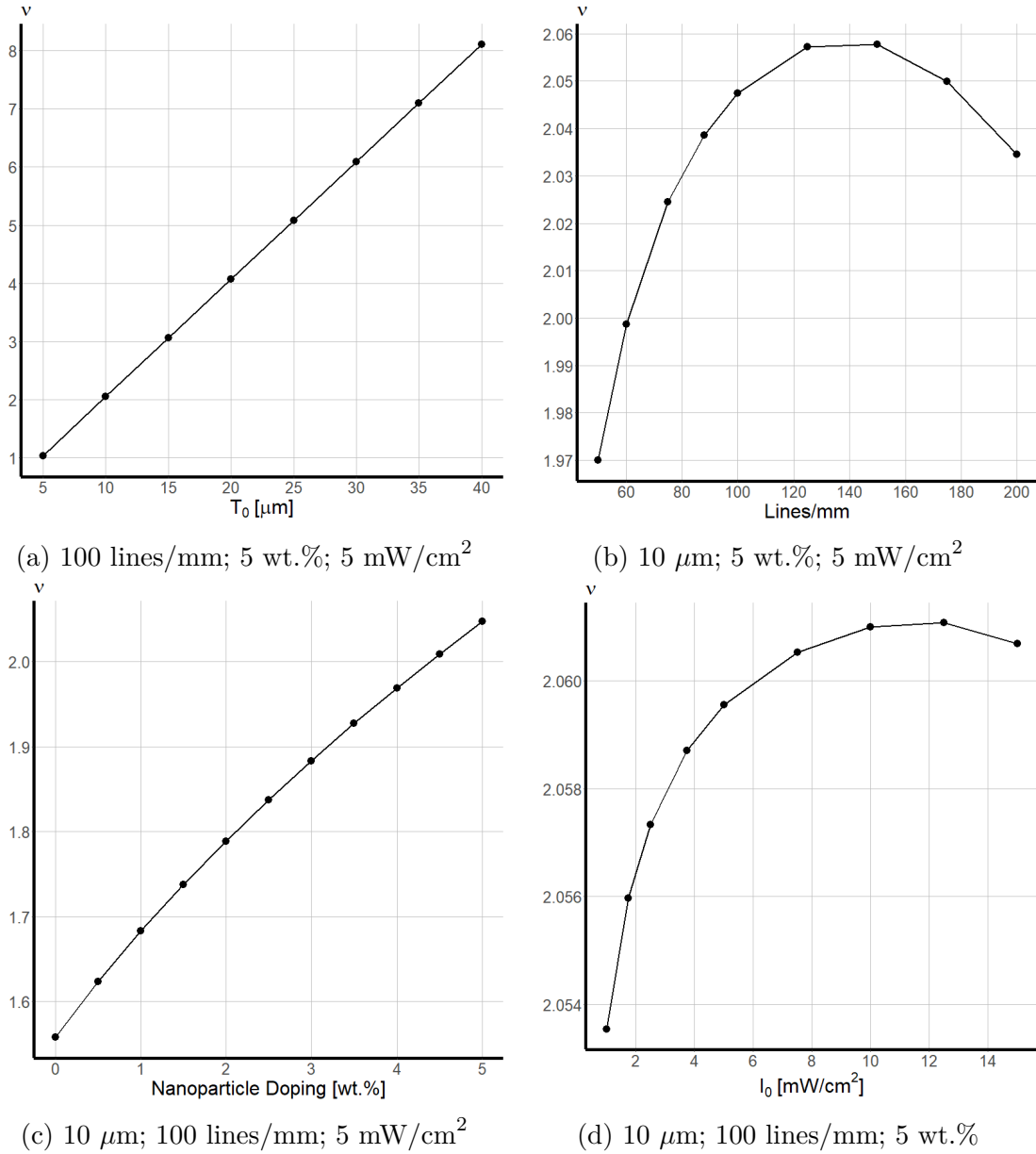
Fig. 4.13 Predicted ν for a variety of different initial conditions.

Alternatively, holographic recording can be terminated early once the desired value of ν has been achieved, see Fig. 4.14.
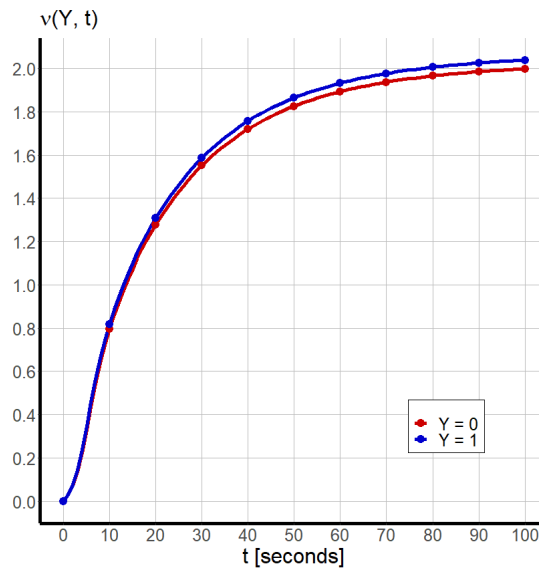
Fig. 4.14 100 lines/mm; 10 μm; 5 wt.%; 0.1 mW/cm^2

The numerical simulation used an absorption coefficient $\zeta = 139$ cm^{-1}, at this value the difference between $\Delta n(y = 0, t)$ and $\Delta n(y = 1, t)$ is very small.

[5] measured the optical properties of an SRG exposed to deionized water and found that diffraction efficiency fell from 20% to 16% within 90 seconds; next, the SRG was also exposed to a solution of Cu^{2+} ions with concentration ranging from 1-4 mM, after 3 minutes diffraction efficiency had fallen to 11.4%, 9.4%, 5% and 3.8% respectively (see Fig. 4.15).
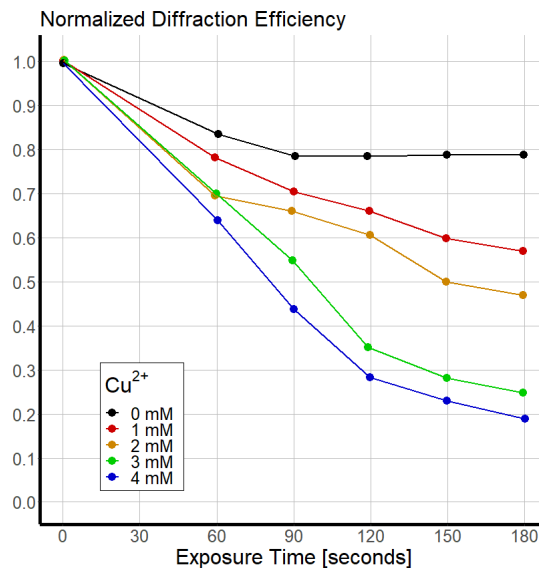
Fig. 4.15 Observed [5] change in diffraction efficiency of an SRG exposed to an aqueous solution with different concentrations of Cu^{2+} ions.

As mentioned earlier, the equations introduced in this work cannot model the formation of an SRG, however, we can predict the overlying trend of these results through numerical simulation of the mathematical model introduced in this chapter. The amount of doping of LTL nanozeolites determines how much η will increase from

its initial value before nanozeolite coating while the question of how much η falls back down is determined by a) the capture and escape coefficients corresponding to the nanozeolites; b) the analyte RI; c) the amount of analyte to which the holographic grating surface is exposed. If $\tau_c^{(s)} < \tau_e^{(s)}$ there will be more vacant nanozeolites replaced with analyte-occupied nanozeolites with a higher RI, $n_z^{(e)} < n_z^{(s)}$; the increased Δn due to the accumulation of low-RI nanozeolites between planes of polymer chains is reduced resulting in a change in diffraction efficiency. Decreasing the escape time means fewer analyte-occupied nanozeolites and more analyte dispersed within the polymer matrix but diffusion of analyte in both the x- and y-directions means there is no concentration gradient of analyte and will not contribute to the RI modulation so the change in diffraction efficiency is less significant. The results presented in Fig. 4.16 show how the model can predict the change in diffraction efficiency of a planar phase grating with $0 \leq a_0/s_0 \leq 0.4$ with four different scenarios; 1) $n_z^{(a)} = 1.5$; 2) $n_z^{(a)} = 1.8$ and weak selectivity, i.e. $\tau_c^{(a)} = \tau_c^{(s)}$, and $\tau_e^{(a)} = \tau_e^{(s)}$; 3) $n_z^{(a)} = 1.8$ and target analyte molecules do not escape from the porous nanoparticles, i.e. the characteristic time for escape of target analyte is infinite $\tau_e^{(a)} = \infty$; 4) $n_z^{(a)} = 1.8$, weak selectivity, $\tau_c^{(a)} = \tau_c^{(s)}$, but the target analyte does not escape the porous nanoparticle $\tau_e^{(a)} = \infty$. The theoretically modelled nanocomposite had an pre-exposure RI modulation and thickness such that $\nu = 1.02$. The following assumptions regarding the secondary analyte were made for the simulation: $\tau_y^{(s)} = 11.6$ s; $\tau_c^{(s)} = 10$ s; $\tau_e^{(s)} = 20$ s; and $n_z^{(s)} = 1.46$.

(a) $n_z^{(a)} = 1.5$

(b) $\tau_c^{(a)} = \tau_c^{(s)}; \tau_e^{(a)} = \tau_e^{(s)}$

(c) $\tau_c^{(a)} < \tau_c^{(s)}; \tau_e^{(a)} = \infty$

(d) $\tau_c^{(a)} = \tau_c^{(s)}; \tau_e^{(a)} = \infty$
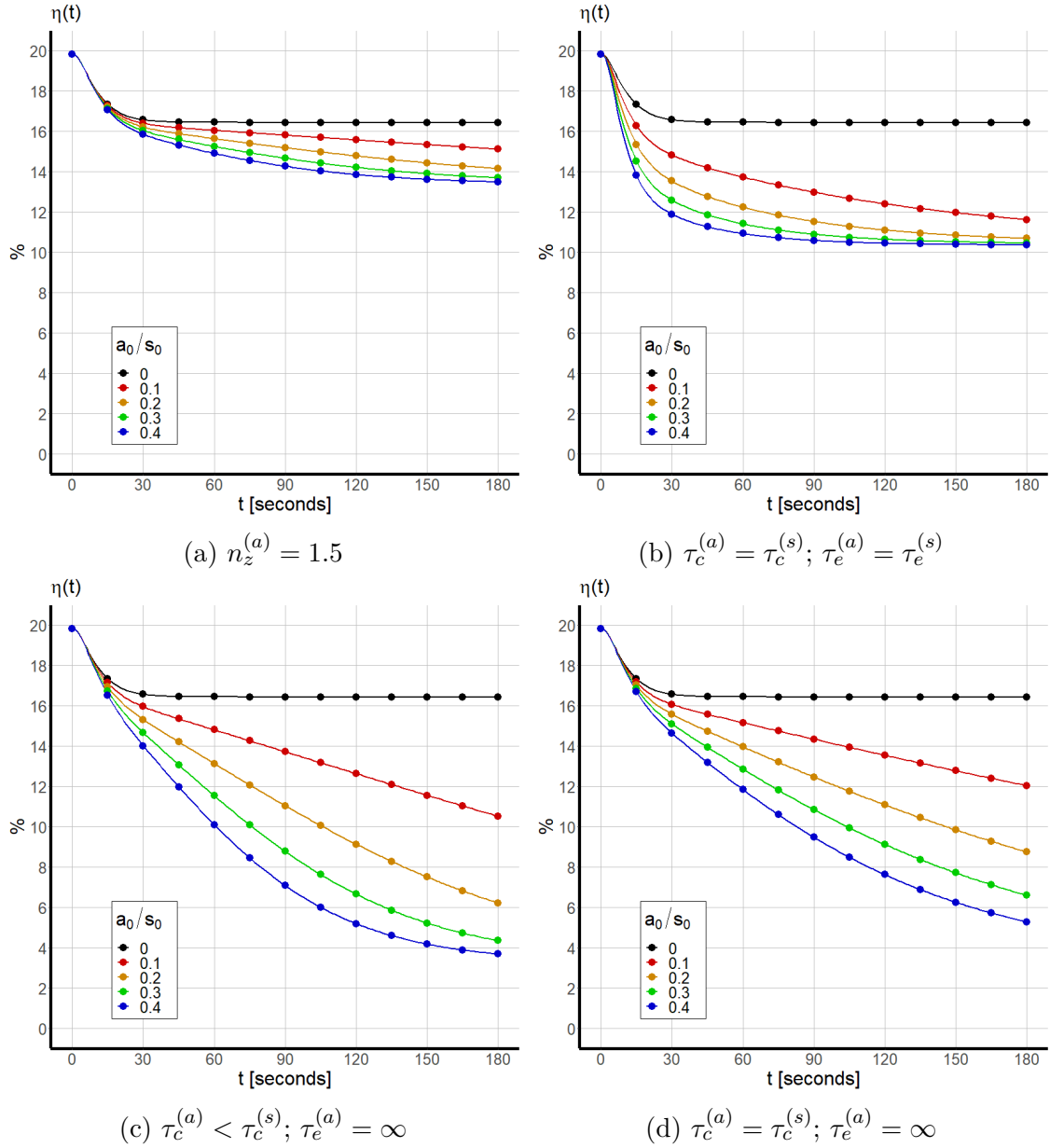
Fig. 4.16 Predicted change in diffraction efficiency of planar phase grating exposed to a target, and secondary, analyte.

One can see that the model is capable of predicting the overlying trend from [5] with the appropriate selection of parameters. The results for $n_z^{(a)} = 1.5$ (Fig. 4.16a) show this value is insufficient, increasing a_0/s_0 produces very little change in RI modulation, and hence, diffraction efficiency. If the characteristic times for capture and escape are independent of the analyte (Fig. 4.16b) there is some change in RI modulation due to the target analyte but changing the concentration does not result in a different change in RI modulation. The predictions for no escape of target analyte (Figs. 4.16c and 4.16d) are reasonably successful at predicted the observed behaviour, increasing the concentration of target analyte results in an increasing change to RI modulation. The characteristic time for the capture of target analyte has minimal effect on the predicted results. These results illustrate the importance

of the ability of the porous nanoparticles to keep target analyte molecules trapped whilst permitting the escape of secondary analytes to the sensor functionality.

4.5   Summary & Conclusions

How can we use the host photopolymer material properties, the recording conditions and the nanodopant properties to control the final grating in a hybrid photopolymer structure and hence optimize its functionality for environmental sensing?

As illustrated in Figs. 4.4 and 4.12 the diffraction efficiency of unslanted holographic gratings is sensitive to changes in RI modulation and thickness, and also, to the initial values for RI modulation and thickness. The equations introduced in Chapters 2 and 3 can model the formation of holographic gratings and can be used to predict the recording conditions and sample preparation that will produce a grating with a RI modulation and thickness that is sensitive to changes in its optical properties and physical dimensions. This chapter introduced a new set of equations to model the change in optical properties of a theoretically modelled holographic grating in response to exposure to a target analyte. The diffusion of a target analyte through a theoretically modelled holographic grating recorded in a hybrid nanocomposite can be modelled by a coupled system of reaction-diffusion equations; the "reaction" terms representing the capture and escape of analyte molecules from porous nanoparticles (Eqn. 4.8). Replacement of vacant nanozeolites with analyte-occupied nanozeolites results in a change in the optical properties of the photonic structure. The new RI spatial profile of the holographic grating $n(x, y, t)$ is calculated with the Lorentz-Lorenz equation. Changes in the optical properties and physical dimensions result in changes to the diffraction efficiency η. For volume phase gratings diffraction efficiency is given by the coupled-wave theory (Eqn. 4.5). The quantity of analyte saturated within the photonic structure can be calibrated from the Bragg angle detuning, after which the characteristic times for the capture and escape of target analyte molecules can also be calibrated from real-time measurements of diffraction efficiency. If the holographic sensor can be calibrated, and if the RI of nanozeolites occupied with a target analyte is known, then the concentration of that target analyte can be estimated via a comparison of experimental results with numerical predictions (Fig. 4.11a). Alternatively, if the concentration of analyte is known then that analyte can be identified by the change in optical properties it produces (Fig. 4.11b). For planar phase gratings (or surface relief gratings) diffraction efficiency is calculated according to Raman-Nath theory (Eqn. 4.12). The model showed some success in predicting the change in diffraction efficiency in a theoretically modelled planar phase grating produced by exposure to a fluid with different concentrations of a known target analyte.

Chapter 5

Future Work

Although the mathematical model presented in this thesis has been mostly successful in achieving what was set out there is some room for improvement. In some instances the model predictions do not reflect the behaviour observed in experimental studies. For example, as stated in the conclusions in Chapter 3 the predicted drop in shrinkage with increased doping of nanozeolites is significantly short of the observed drop [75, 59]; furthermore, the predicted drop in shrinkage with increased initial thickness of the sample is also shy of the observed drop [72]. This chapter will present a few conjectures that could be investigated with the view to improving the mathematical model.

5.1 Modelling the Redistribution of Photoinitiator, Sensitizing Dye and Cross-linker

The photopolymer system consists of a monomer, a cross-linking monomer, a sensitizing dye and a photoinitiator dispersed within a binder matrix, but in modelling the photochemical and diffusion processes in this system the first four constituents are treated as a single entity. This assumption works reasonably well for modelling holographic recording in materials with a fixed formula for the relative quantities, but some of the earlier mathematical models for the formation of holographic gratings [33] believed that to effectively model holographic recording in a generalized undoped photopolymer system a more realistic model is needed which should treat the concentration of each constituent separately. If we denote the mass concentrations of the monomer m_1, the cross-linking monomer m_2, the photoinitiator i, the sensitizing dye d, short polymers p, long cross-linked polymers q and inorganic nanoparticles z then the photochemical reaction resulting in the formation of mobile short polymer is $\Phi(t)F(x)idm_1$ and the formation of immobile cross-linked polymer chains is $\Phi(t)\Gamma(x)idm_2p^2$. Hence, the extended system of PDEs governing the evolution of the mass concentrations of each of the constituents will be

$$\frac{\partial m_1}{\partial t} + \vec{\nabla} \cdot J_{m_1} = -\Phi(t)F(x)idm_1,$$

$$\frac{\partial m_2}{\partial t} + \vec{\nabla} \cdot J_{m_2} = -\Phi(t)\Gamma(x)idm_2p^2,$$

$$\frac{\partial i}{\partial t} + \vec{\nabla} \cdot J_i = -\Phi(t)F(x)idm_1 - \Phi(t)\Gamma(x)idm_2p^2,$$

$$\frac{\partial d}{\partial t} + \vec{\nabla} \cdot J_d = -\Phi(t)F(x)idm_1 - \Phi(t)\Gamma(x)idm_2p^2,$$

$$\frac{\partial p}{\partial t} + \vec{\nabla} \cdot J_p = \Phi(t)F(x)idm_1 - \Phi(t)\Gamma(x)idm_2p^2,$$

$$\frac{\partial q}{\partial t} = \Phi(t)\Gamma(x)idm_2p^2,$$

$$\frac{\partial z}{\partial t} + \vec{\nabla} \cdot J_z = 0.$$

Note also that the cross-linking is not a constant but rather a function of x. There is empirical evidence [7] from real-time measurements of RI modulation over short exposures in photopolymer media of a time delay between the end of recording and the termination of polymerization at low spatial frequencies. One conjecture for this phenomenon is that cross-linking of short polymer chains does not require illumination and the $\Phi(t)\Gamma p^2$ in Eqn. 3.3 can be replaced with Γp^2. Alternatively, if illumination is required for cross-linking then the probability for growth of immobile polymer chains should be higher at the bright fringes of the illumination pattern in which case the rate of immobilization should be multiplied by $f(x)$. Hence, the cross-linking reaction term could be expressed as $\Phi(t)\Gamma_0 f(x)idm_2p^2$. The relationship between cross-linking short polymers and the intensity of the recording pattern will require further experimental and theoretical work.

5.2   Rotating Domain for the Formation of Slanted Gratings

One possible explanation for the deficiencies of the model's predicted shrinkage lies with how the spatial domain is defined. If the axes are defined as in Fig. 5.1. The domain for our model is

$$0 \leq x \leq \hat{x}, \qquad\qquad 0 \leq w \leq \frac{T(t)}{\cos\phi_r(t)}, \qquad\qquad t \geq 0.$$
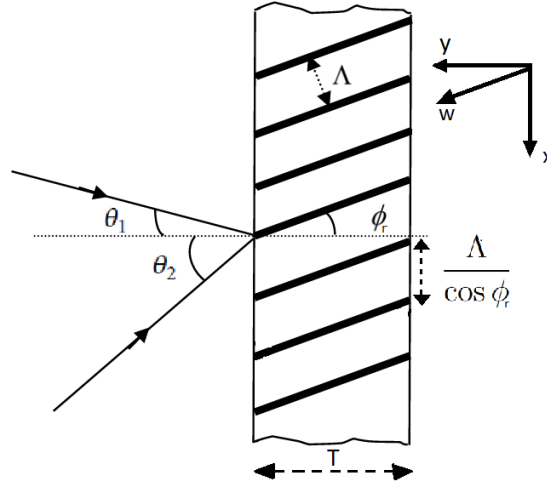
Fig. 5.1 A rotating spatial domain

The redistribution of monomer, polymer and inorganic nanoparticles can still be modelled by Eqn. 3.3, however the gradient vector will be

$$\nabla = \frac{\partial}{\partial x}\mathbf{e_x} + \sec\phi_r(t)\frac{\partial}{\partial w}\mathbf{e_w},$$

where $\mathbf{e_x}$ and $\mathbf{e_w}$ are unit vectors in the x and w directions respectively. Also, the flux vectors become

$$\vec{J_m} = - D_m\frac{\partial m}{\partial x}\mathbf{e_x} - D_m\sec\phi_r(t)\frac{\partial m}{\partial w}\mathbf{e_w},$$

$$\vec{J_p} = - D_p\left\{\left[\frac{\partial p}{\partial x}\mathbf{e_x} + \sec\phi_r(t)\frac{\partial p}{\partial w}\mathbf{e_w}\right] + \epsilon_z\left[\frac{\partial(pz)}{\partial x}\mathbf{e_x} + \sec\phi_r(t)\frac{\partial(pz)}{\partial w}\mathbf{e_w}\right]\right\},$$

$$\vec{J_z} = - D_z\left\{\left[\frac{\partial z}{\partial x}\mathbf{e_x} + \sec\phi_r(t)\frac{\partial z}{\partial w}\mathbf{e_w}\right] + \epsilon_z\left[\frac{\partial(pz)}{\partial x}\mathbf{e_x} + \sec\phi_r(t)\frac{\partial(pz)}{\partial w}\mathbf{e_w}\right] + \right.$$
$$\left. \epsilon_z\left[\frac{\partial(qz)}{\partial x}\mathbf{e_x} + \sec\phi_r(t)\frac{\partial(qz)}{\partial w}\mathbf{e_w}\right]\right\}.$$

Note that $\mathbf{e_w}$ is time dependent and will rotate with the Bragg planes. The feasibility of this model using a rotating unit vector will require further theoretical investigation.

5.3   Bending Bragg Planes

Absorption of light within the holographic material means a higher concentration of high-RI immobilized polymer chains near the free surface than at the fixed surface; the resulting concentration gradient drives cross-diffusion of inorganic nanoparticle and, consequently, an accumulation of low-RI inorganic nanoparticles at the fixed surface. Consequently, the average RI of the holographic grating drops from $Y = 1$ to $Y = 0$. The depth variation of mean RI in a holographic grating means that an optical wavefront is progressively refracted as it propagates through the material. The fringe plane rotation model requires the parallel planes of polymer chains to

remain unbent throughout holographic recording, i.e. that $\phi_r \neq \phi_r(Y, t)$. However, the probe beam is not subject to the constraint and will bend as it propagates through the photonic structure. This phenomenon is illustrated in Fig. 5.2, the purple line is how the probe beam would propagate in a material with $n_2(Y, t) = n_2 \approx n_1$, the slant will be as predicted by solutions to Eqn. 3.19a; the green line shows how the wavefront should look if $n_2(Y, t) = n_2 > n_1$, the high RI of the holographic material refracts the wavefront towards the normal; the red curve is how the wavefront will look if $\partial n_2 / \partial Y > 0$, the greater the difference between $n_2(Y, t)$ and n_1, the more the probe beam will refract towards the normal; the blue curve shows how the wavefront bends away from the normal if $\partial n_2 / \partial Y < 0$.
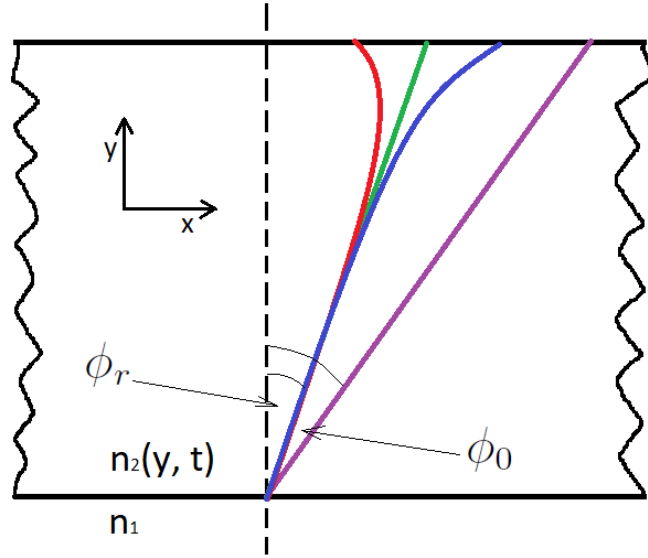
Fig. 5.2 Bending of the probe beam.

Curvature of probe beam can cause side-lobe asymmetries in the angular sensitivity curve, this effect becomes more noticeable throughout holographic recording as n_2 changes with time. Polymerization induced shrinkage in holographic gratings recorded on hybrid photopolymer materials is measured, indirectly, by the detuning of the Bragg angle (Eqn. 3.20). This calculation is based on the fringe rotation model [87], however, the conversion of monomer to a higher RI polymer causes a change in the mean RI which in turn produces a variation in the reconstruction geometry for maximum diffraction efficiency, in other words, a detuning of the Bragg angle. For an optical wavefront passing from a material with RI n_1 to a material with n_2 at an angle of incidence θ_i, the well known Snell's law of refraction

$$n_1 \sin \theta_i = n_2 \sin \theta_p, \tag{5.2}$$

will govern the angle θ_p of the probe beam within the holographic grating. If the mean RI of the holographic material is a function of time and the depth within the

material, then the same must be true of the angle of refraction

$$n_2 = n_2(Y, t), \qquad\qquad \theta_p = \theta_p(Y, t).$$

If $\overline{n}(Y, t)$ is the average RI across the holographic grating and assuming that at the point of incidence the probe beam is parallel to the Bragg planes, in other words that $\theta_p(0, t) = \phi_r(t)$, then Snell's law gives us

$$n_1 \sin \theta_i = \overline{n}(Y, t) \sin \theta_p(Y, t) = \overline{n}(0, t) \sin \phi_r(t) = \overline{n}(1, t) \sin \theta_p(1, t).$$

The consequence of Bragg plane bending is an additional Bragg angle detuning and hence a contribution to apparent shrinkage. Further investigation into this phenomenon could elucidate the drop in relative shrinkage with increased doping of nanozeolites or increased thickness.

5.4 Change in Physical Dimensions of a Holographic Sensor

In Chapter 4 we assumed that the change in thickness resulting from the influx of fluid is negligible and, consequently, the y domain is fixed. If one drops that assumption then

$$0 \leq y \leq T_2(t),$$

where T_2 is the thickness of the nanocomposite after the influx of analyte material. If T_1 is the pre-exposure thickness then the pre-exposure volume of the holographic grating relative to m_0 prior to the analyte exposure is

$$v_1 = \frac{b_0}{\rho_b} + \frac{z_0}{\rho_z} + \frac{1}{\hat{x} T_1} \int_0^{\hat{x}} \int_0^{T_1} \left[\frac{m(x, y)}{\rho_m} + \frac{p(x, y)}{\rho_p} + \frac{q(x, y)}{\rho_p} \right] \, dx \, dy,$$

where m, p and q are the final mass distributions from modelling the formation of the slanted grating, i.e. the final condition from Eqn. 3.3. If the density of the target analyte and secondary analyte are ρ_a and ρ_s, then the volume of the holographic grating after the analyte has entered the photonic structure will be

$$\begin{aligned}
v_2(t) = \frac{b_0}{\rho_b} + \frac{z_0}{\rho_z} + \frac{1}{\hat{x} T_2} \int_0^{\hat{x}} \int_0^{T_2} &\left[\frac{m(x, y)}{\rho_m} + \frac{p(x, y)}{\rho_p} + \frac{q(x, y)}{\rho_p} \right.\\
&\left. + \frac{s(x, y, t)}{\rho_s} + \frac{a(x, y, t)}{\rho_a} \right] \, dx \, dy.
\end{aligned}$$

There is an increase in volume after the analyte enters the photonic structure, however, as analyte molecules are captured and trapped within the interior of the porous nanozeolites the expanded volume will deflate. After all available vacant nanozeolites have been replaced with occupied nanozeolites any remaining uncaptured analyte

molecules will contribute to a net increase in volume. If $\phi_{r,1}$, Λ_1 and T_1 are the slant angle inside the grating, grating period and the thickness respectively, and if the back of the grating is fixed to a rigid substrate throughout the exposure to the analyte then the same restrictions (Eqn. 3.1) which applied to modelling the formation of the slanted grating must also apply to modelling the response of the holographic sensor. We will assume that all change in the physical dimensions of the holographic grating takes place in the thickness. Hence, the new thickness of the holographic grating is

$$T_2(t) = T_1 \frac{v_2(t)}{v_1}. \tag{5.3}$$

If the slant angle is non-zero the change in thickness will result in a detuning of the Bragg angle. The equations describing the time evolution of slant angle (Eqn. 3.19a) and grating period (Eqn. 3.19b) apply to modelling the response of a holographic sensor

$$\phi_{r,2}(t) = \tan^{-1}\left(\frac{T_1 \tan \phi_{r,1}}{T_2(t)}\right), \tag{5.4}$$

$$\Lambda_2(t) = \Lambda_1 \frac{\cos \phi_{r,2}(t)}{\cos \phi_{r,1}}. \tag{5.5}$$

Furthermore, if there is a transfer of target analyte and secondary analyte across the boundary from outside the domain to within then new boundary conditions need to be defined. As in Chapter 3 boundary immobilization methods must be employed to resolve the time dependent domain

$$Y = \frac{y}{u(t)}, \qquad a(x,y,t) = A(x,Y,t), \qquad s(x,y,t) = S(x,Y,t),$$

$$z^{(e)}(x,y,t) = Z^{(e)}(x,Y,t), \quad z^{(a)}(x,y,t) = Z^{(a)}(x,Y,t), \quad z^{(s)}(x,y,t) = Z^{(s)}(x,Y,t),$$

$$b(x,y,t) = B(x,Y,t), \qquad m(x,y,t) = M(x,Y,t), \qquad p(x,y,t) = P(x,Y,t),$$

$$q(x,y,t) = Q(x,Y,t).$$

Applying the transformation to Eqn. 4.11

$$\frac{\partial A}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial A}{\partial Y} + \alpha_{a,x}\frac{\partial^2 A}{\partial x^2} + \frac{\alpha_{a,y}}{u^2}\frac{\partial^2 A}{\partial w^2} - \gamma_{a,a}AZ^{(e)} + \omega_{a,a}Z^{(a)}, \tag{5.6a}$$

$$\frac{\partial S}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial S}{\partial Y} + \alpha_{s,x}\frac{\partial^2 S}{\partial x^2} + \frac{\alpha_{s,y}}{u^2}\frac{\partial^2 S}{\partial w^2} - \gamma_{s,s}SZ^{(e)} + \omega_{s,s}Z^{(s)}, \tag{5.6b}$$

$$\frac{\partial Z^{(e)}}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial Z^{(e)}}{\partial Y} - \gamma_{a,z}AZ^{(e)} + \omega_{a,z}Z^{(a)} - \gamma_{s,z}SZ^{(e)} + \omega_{s,z}Z^{(s)}, \tag{5.6c}$$

$$\frac{\partial Z^{(a)}}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial Z^{(a)}}{\partial Y} + \gamma_{a,z}AZ^{(e)} - \omega_{a,z}Z^{(a)}, \tag{5.6d}$$

$$\frac{\partial Z^{(s)}}{\partial t} = \frac{Y}{u}\frac{du}{dt}\frac{\partial Z^{(s)}}{\partial Y} + \gamma_{s,z}SZ^{(e)} - \omega_{s,z}Z^{(s)}. \tag{5.6e}$$

The fixed domain is

$$0 \leq x \leq 1, \qquad\qquad 0 \leq Y \leq 1, \qquad\qquad t \geq 0.$$

In addition the transformation requires equation for the distribution of binder, monomer, short polymer and cross-linked polymer

$$\frac{\partial B}{\partial t} = \frac{Y}{u} \frac{du}{dt} \frac{\partial B}{\partial Y},$$
$$\frac{\partial M}{\partial t} = \frac{Y}{u} \frac{du}{dt} \frac{\partial M}{\partial Y},$$
$$\frac{\partial P}{\partial t} = \frac{Y}{u} \frac{du}{dt} \frac{\partial P}{\partial Y},$$
$$\frac{\partial Q}{\partial t} = \frac{Y}{u} \frac{du}{dt} \frac{\partial Q}{\partial Y}.$$

The boundary conditions are as follows,

$$\frac{\partial^n A}{\partial x^n}(0, Y, t) = \frac{\partial^n A}{\partial x^n}(1, Y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$
$$\frac{\partial A}{\partial Y}(x, 0, t) = \frac{\partial A}{\partial Y}(x, T, t) = 0,$$
$$A(x, 1, t) = 1,$$

$$\frac{\partial^n A}{\partial x^n}(0, Y, t) = \frac{\partial^n S}{\partial x^n}(1, Y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$
$$\frac{\partial S}{\partial Y}(x, 0, t) = \frac{\partial S}{\partial Y}(x, T, t) = 0,$$
$$S(x, 1, t) = 1,$$

$$\frac{\partial^n Z^{(e)}}{\partial x^n}(0, Y, t) = \frac{\partial^n Z^{(e)}}{\partial x^n}(1, Y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$
$$\frac{\partial Z^{(e)}}{\partial Y}(x, 0, t) = \frac{\partial Z^{(e)}}{\partial Y}(x, 1, t) = 0,$$

$$\frac{\partial^n Z^{(a)}}{\partial x^n}(0, Y, t) = \frac{\partial^n Z^{(a)}}{\partial x^n}(1, Y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$
$$\frac{\partial Z^{(a)}}{\partial Y}(x, 0, t) = \frac{\partial Z^{(a)}}{\partial Y}(x, 1, t) = 0,$$

$$\frac{\partial^n Z^{(s)}}{\partial x^n}(0, Y, t) = \frac{\partial^n Z^{(s)}}{\partial x^n}(1, Y, t) \qquad\qquad n = \{0, 1, 2, ...\},$$
$$\frac{\partial Z^{(s)}}{\partial Y}(x, 0, t) = \frac{\partial Z^{(s)}}{\partial Y}(x, 1, t) = 0,$$

$$\frac{\partial B}{\partial Y}(x,0,t) = \frac{\partial B}{\partial Y}(x,1,t) = 0,$$

$$\frac{\partial M}{\partial Y}(x,0,t) = \frac{\partial M}{\partial Y}(x,1,t) = 0,$$

$$\frac{\partial P}{\partial Y}(x,0,t) = \frac{\partial P}{\partial Y}(x,1,t) = 0,$$

$$\frac{\partial Q}{\partial Y}(x,0,t) = \frac{\partial Q}{\partial Y}(x,1,t) = 0.$$

Further research into this model should lead to the establishment of a robust mathematical description of the response of a volume phase holographic grating in response to exposure to a fluid containing multiple analytes.

5.5 Surface Relief Gratings

In Chapter 4 we briefly mentioned how the equations from chapters 2 and 3 are not capable of modelling the formation of SRGs due to the post-recording thermal treatment and spin coating processes. To reiterate, SRGs are fabricated by 1) recording a planar phase holographic grating onto an undoped photopolymer material; 2) the holographic grating is placed in an oven for thermal treatment whereby all bulk material is evaporated leaving behind only cross-linked polymer, the photonic structure is just a few hundred nanometers thick; 3) the holographic grating is spin coated with nanozeolites incorporated within a sol gel layer. This first stage can be modelled with Eqn. 3.3 with $z_0 = 0$, the final condition of Eqn. 3.3c gives us the mass concentration of cross-linked polymer as a function of x and y. The thermal treatment process could be modelled by simply removing all components other than the cross-linked polymer, i.e. the nanocomposite becomes a one-component system $q(x,y)$. The concentration of polymer is a periodic function, $q(x+\Lambda,y) = q(x,y)$, and can be represented by a Fourier series

$$q(x,y) \approx \sum_{i=0} Q_i^{(a)}(y) \cos\left(\frac{2\pi}{\Lambda}ix\right) + Q_i^{(b)}(y) \sin\left(\frac{2\pi}{\Lambda}ix\right). \tag{5.7}$$

After the spin coating is complete the nanozeolites fill in the troughs between the polymer structures, the distribution of nanozeolites can be modelled as mirror image of the spatial profile of the cross-linked polymer, hence the distribution of nanozeolites can be modelled by the first harmonics

$$z(x,y) = \sum_{i=0} Q_i^{(a)}\left[1 - \cos\left(\frac{2\pi}{\Lambda}ix\right)\right] + Q_i^{(b)}\left[1 - \sin\left(\frac{2\pi}{\Lambda}ix\right)\right]. \tag{5.8}$$

If T_0 is the pre-thermal treatment thickness of the photonic structure and one assumes that the density of the cross-linked polymer does not change after thermal treatment

then the thickness after thermal treatment and spin coating can be expressed as

$$T_{SRG} = \frac{T_0}{m_0} \left(\frac{\rho_p}{\rho_m} + \frac{b_0}{m_0} \frac{\rho_p}{\rho_b} \right)^{-1} \int_0^{T_0} \int_0^{\Lambda} q(x,y) \ dx \ dy. \tag{5.9}$$

The predicted distribution of cross-linked polymer and nanozeolites from numerical simulation of the formation of an SRG recorded with spatial frequency of 300 lines/mm on a 30 μm thick AA/PVA sample (50:1 binder-to-monomer) is presented in Fig. 5.3.

(a) Cross-linked polymer   (b) Nanozeolites
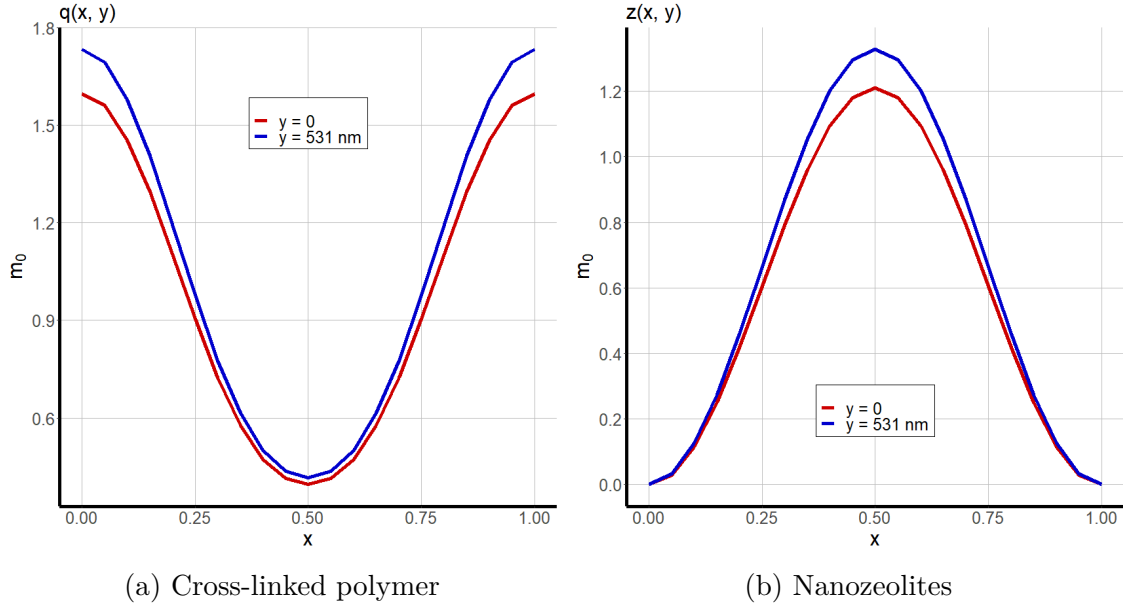
Fig. 5.3 Distribution of cross-linked polymer and nanozeolites in theoretically modelled SRG after thermal treatment and spin coating.

The RI of the SRG is calculated via the Lorentz-Lorenz equation as before, however now there are only two components

$$\frac{n^2 - 1}{n^2 + 2} = \phi_q \frac{n_q^2 - 1}{n_q^2 + 2} + \phi_z \frac{n_z^2 - 1}{n_z^2 + 2}.$$

From this point on the SRG can be modelled the same as any other planar phase grating.

The changes suggested in this chapter need to be justified by comparing the predicted results with existing experimental data. This will require further experimental and theoretical research.

Chapter 6

Final Conclusions

The motivation for this research project was to develop a set of mathematical models that will characterize, and ultimately guide the design of, photosensitive materials capable of copying with high fidelity an illumination pattern into a holographic grating. Prior to this project our collaborators already achieved significant progress with this problem in the context of pure organic photopolymer systems. This project aimed firstly to expand our study to hybrid materials containing nanozeolites, addressing issues such as shrinkage minimization, high-fidelity copying and optimization of the nanoparticle redistribution. In addition, we wanted to use these models for establishing a theoretical framework for the fabrication of biomedical and environmental monitoring, zeolite-based holographic sensors. The primary objectives were

- to extend the existing models to hybrid photopolymer systems containing inorganic nanoparticle and to quantify how these nanoparticles affect, and are redistributed during, a holographic recording;

- study the effect of cross-diffusion of photopolymer components on grating formation models;

- to incorporate two spatial variables into existing models in order to study depth effects such as slanted gratings, shrinkage during photopolymerization;

- model the time response of holographic sensors based on the diffusion dynamics of a specific target analyte and the structure of the sensor (operating in transmission or reflection mode).

We began, in Chapter 2, with the assumption that the redistribution of inorganic nanoparticles in the recording of an unslanted holographic grating in a hybrid photopolymer material can be modelled by a one-dimensional diffusion equation and that the addition of inorganic nanoparticles to the photopolymer matrix results in photon scattering which reduces the visibility of the recorded illumination pattern. Redistribution of nanoparticles is driven by a combination of self-diffusion and mutual cross-diffusion between inorganic nanoparticles and both mobile and immobile polymer. Hence, it was necessary to solve the nanoparticle diffusion equation

simultaneously with the PDEs modelling the formation and distribution of immobile polymer chains. Solutions to this diffusion equation can be used to model the fraction of inorganic nanoparticles that are redistributed in holographic recording. If the densities and RI of the binder, monomer, short polymer, cross-linked polymer and inorganic nanoparticle are known, then solutions to the coupled system of PDEs can used with the Lorentz-Lorenz equation to model the grating profile of the nanocomposite's RI. The mass concentrations of each part of the nanocomposite are defined on a periodic spatial domain. Hence, the RI can be modelled as a Fourier expansion series with the first harmonic a good approximation for the RI modulation and the higher order harmonics acting as a measure of the distortion of the holographic grating from the illumination pattern. Through extensive numerical simulation of the model with a variety of different values for cross-diffusion and scattering coefficients experimental results could be accurately predicted. The combination of mutual cross-diffusion and photon scattering results in an optimal doping beyond which there is a disimprovement in performance with increased doping. Numerical simulation of the mathematical model found optimal doping between 1-4 wt.% with the optimal value increasing with increased spatial frequency, also that low RI nanoparticle perform better in terms of high RI modulation and low distortion.

In Chapter 3 we found that the one dimensional equations were insufficient for modelling polymerization induced shrinkage and it was deemed necessary to the extend the equations to a second spatial dimension to accommodate the non-zero slant angle and absorption of light across the finite depth of the photonic structure. By extending the diffusion flux expressions to a direction parallel to the depth of the sample the temporal evolution and spatial redistribution of monomer, short polymer, cross-linked polymer and inorganic nanoparticles can be described by a system of reaction-diffusion equations. A new polymerization function was needed to model the drop in intensity and the shifting illumination pattern within the depth of the nanocomposite. By averaging the mass concentrations over the spatial domain it was shown how the volume of the grating could be expressed as a function of time, these expressions can be used to model polymerization induced shrinkage and the Bragg angle detuning. Due to the changes in RI and grating period inside the nanocomposite during holographic recording there is an additional Bragg angle detuning resulting in an apparent shrinkage which will differ from the actual shrinkage. Numerical simulation of the model found that the actual shrinkage is largely determined by the mass ratio of binder to monomer; the difference density between the polymer and monomer; the recording intensity (for a fixed energy input); the absorption coefficient; and the fraction of the initial quantity of monomer that has been polymerized. Furthermore, the apparent shrinkage is determined by the RI of the polymer, the binder and, to a lesser extent, the RI of the nanoparticles. The spatial frequency and slant angle also have a substantial impact on apparent shrinkage. As in the case of the one-dimensional analysis, the RI of a slanted holographic grating can be modelled by a Fourier series. However, the new spatial domain means the sine

harmonics are non-zero, modelling RI modulation was achieved by calculating the norm of the vector whose coordinates are the first order harmonics. Distortion, that is to say, the extent to which the theoretically modelled holographic grating deviates from the illumination pattern could still be modelled by the higher order harmonics. Numerical simulation of the model found that RI modulation is at its maximum within the interior of the photonic structure. By increasing the initial thickness of the sample the model predicted increased RI modulation at the surface exposed to holographic recording and reduced RI modulation at the opposite surface. The model also predicted lower distortion at the fixed surface and greater distortion at the free surface with increased thickness. The initial slant angle had no effect on the neither the predicted RI modulation nor distortion. By modelling RI modulation and the average RI the geometry of the grating can be predicted via the Moharam-Young criteria.

As illustrated in Figs. 4.4 and 4.12 the diffraction efficiency of unslanted holographic gratings is sensitive to changes in RI modulation and thickness, and also, to the initial values for RI modulation and thickness. The results presented in Figs. 4.5 and 4.13 show how numerical simulation of the mathematical model introduced in Chapters 2 and 3 can offer some insight into the recording conditions and sample preparation which will result in the formation of a holographic grating with a RI modulation and thickness such that its diffraction efficiency is most sensitive to changes in optical properties or changes in physical dimensions. Chapter 4 introduced a new set of equations to model the change in optical properties in response to exposure to a solution of a target analyte. The initial condition of the nanocomposite was modelled using the final conditions from the equations introduced in chapters 2 and 3. The diffusion of a target analyte and solvent through a theoretically modelled holographic grating recorded in a hybrid nanocomposite can be modelled by a coupled system of reaction-diffusion equations; the "reaction" terms representing the capture and escape of analyte or solvent molecules from porous nanoparticles. As the solution penetrates into the depth of the nanocomposite vacant nanozeolites are replaced with solvent-occupied and analyte-occupied nanozeolites which results in a change in the optical properties of the photonic structure. As in previous chapters the Lorentz-Lorenz equation is used to model the new RI spatial profile of the holographic grating. Changes in the optical properties of the nanocomposite result in changes to the diffraction efficiency η. For volume phase gratings diffraction efficiency is given by the coupled-wave theory, numerical simulation of the model showed how the quantity of solvent saturated within the photonic structure can be inferred from the Bragg angle detuning. The characteristic times for the capture and escape of solvent and target analyte molecules can be calibrated from real-time measurements of diffraction efficiency. If the holographic sensor can be calibrated, and if the RI of nanozeolites occupied with a target analyte is known, then the concentration of that target analyte can be estimated via a comparison of experimental results with numerical predictions. Alternatively, if the concentration of analyte in a solution

is known then that analyte can be identified by the change in optical properties it produces. For planar phase gratings (or surface relief gratings) diffraction efficiency is calculated according to Raman-Nath theory. The model proved somewhat successful in predicting the change in diffraction efficiency in a theoretically modelled planar phase grating produced by exposure to different concentrations of a target analyte.

This thesis has provided a mathematical framework capable of modelling a) the formation of holographic gratings in hybrid nanocomposite materials with good accuracy, and b) the changes in optical properties of a theoretically modelled holographic sensor in response to exposure to a target analyte. Our model can offer practical insights and guide decision making for experimental scientists and help identify opportunities for improvement for our collaborators. We believe this work could be improved in future research by investigating the change in physical dimensions of a theoretically modelled volume phase grating resulting from the influx of a miscible fluid and extending the model to surface relief gratings.

References

[1] Tatsiana Mikulchyk. *Development of holographic sensors for monitoring relative humidity and temperature*. PhD thesis, Technological University Dublin, 2016.

[2] P. O'Reilly. *Mathematical Modelling of Optical Patterning in Photopolymer Systems*. PhD thesis, Technological University Dublin, 2016.

[3] P. Blanche, editor. *Optical Holography: Materials, Theory and Applications*. Elsevier, 2020.

[4] M. Moothanchery. *Studies of shrinkage in photopolymerisable materials for holographic applications*. PhD thesis, Technological University Dublin, 2013.

[5] S. Gul, D. Cody, A. Kharchenko, S. Martin, S. Mintova, J. Cassidy, and I. Naydenova. Ltl type nanozeolites utilized in surface photonics structures for environmental sensors. *Microporous and Mesoporous Materials*, 261:268–274, 2018.

[6] I. Naydenova, E. Leite, T. Babeva, N. Pandey, T. Baron, T. Yovcheva, S. Sainov, S. Martin, S. Mintova, and V. Toal. Optical properties of photopolymerizable nanocomposites containing nanosized molecular sieves. *Journal of Optics*, 13(4):044019, March 2011.

[7] T. Babeva, I. Naydenova, D. Mackey, S. Martin, and V. Toal. Two-way diffusion model for short-exposure holographic grating formation in acrylamide-based photopolymer. *JOSA B*, 27(2):197–203, 2010.

[8] T. Babeva, D. Mackey, I. Naydenova, S. Martin, and V. Toal. Study of the photoinduced surface relief modulation in photopolymers caused by illumination with a gaussian beam of light. *Journal of Optics*, 12(12):124011, 2010.

[9] Martin J Klein. Max planck and the beginnings of the quantum theory. *Archive for History of Exact Sciences*, 1:459–479, 1961.

[10] Albert Einstein. Quantum theory of the monatomic ideal gas. *Sitzungsberichte der Preussischen Akademie der Wissenschaften, Physikalisch-mathematische Klasse*, 261, 1924.

[11] H. Kogelnik. Coupled wave theory for thick hologram gratings. In *Landmark Papers On Photorefractive Nonlinear Optics*, pages 133–171. World Scientific, 1995.

[12] C. V. Raman and N. Nath. The diffraction of light by high frequency sound waves: Part i. In *Proceedings of the Indian Academy of Sciences-Section A*, volume 2, pages 406–412. Springer, 1935.

[13] C. V. Raman and N. Nath. The diffraction of light by sound waves of high frequency: Part ii. In *Proceedings of the Indian Academy of Sciences-Section A*, volume 2, pages 413–420. Springer, 1935.

[14] C. V. Raman and N. Nath. The diffraction of light by high frequency sound waves: Part iii. In *Proceedings of the Indian Academy of Sciences-Section A*, volume 3, pages 75–84. Springer, 1936.

[15] W. R. Klein and B. D. Cook. Unified approach to ultrasonic light diffraction. *IEEE Transactions on Sonics and Ultrasonics*, 14(3):123–134, 1967.

[16] M. G. Moharam and L. Young. Criterion for bragg and raman-nath diffraction regimes. *Applied Optics*, 17(11):1757–1759, 1978.

[17] H. I. Bjelkhagen. *Silver-halide recording materials for holography and their processing*, volume 66. Springer Science & Business Media, 1995.

[18] C. Solano, R. A. Lessard, and P. C. Roberge. Methylene blue sensitized gelatin as a photosensitive medium for conventional and polarizing holography. *Applied Optics*, 26(10):1989–1997, 1987.

[19] A. del Campo and C. Greiner. Su-8: a photoresist for high-aspect-ratio and 3d submicron lithography. *Journal of micromechanics and microengineering*, 17(6):R81, 2007.

[20] L. B. Glebov. Photochromic and photo-thermo-refractive glasses. *Encyclopedia of Smart Materials*, 2002.

[21] M. R. Gleeson, D. Sabol, S. Liu, C. E. Close, J. V. Kelly, and J. T. Sheridan. Improvement of the spatial frequency response of photopolymer materials by modifying polymer chain length. *JOSA B*, 25(3):396–406, 2008.

[22] E. Leite, I. Naydenova, N. Pandey, T. Babeva, G. Majano, and S. Mintova. Investigation of the light induced redistribution of zeolite beta nanoparticles in an acrylamide-based photopolymer. *Journal of Optics: Pure and Applied Optics*, 11(2), 2009.

[23] I. Naydenova, R. Jallapuram, V. Toal, and S. Martin. Characterisation of the humidity and temperature responses of a reflection hologram recorded in acrylamide-based photopolymer. *Sensors and Actuators B: Chemical*, 139(1):35–38, 2009.

[24] Emma Branigan, Suzanne Martin, Matthew Sheehan, and Kevin Murphy. Analog holographic wavefront sensor for defocus and spherical aberration measurement recorded in a photopolymer. *Opt. Express*, 31(6):9466–9480, March 2023.

[25] V. K. Vanag and I. R. Epstein. Cross-diffusion and pattern formation in reaction–diffusion systems. *Physical Chemistry Chemical Physics*, 11(6):897–912, 2009.

[26] G. Zhao and P. Mouroulis. Diffusion model of hologram formation in dry photopolymer materials. *Journal of Modern Optics*, 41(10):1929–1939, 1994.

[27] S. Piazzola and B. Jenkins. First-harmonic diffusion model for holographic grating formation in photopolymers. *J. Opt. Soc. Am. B*, 17:1147–1157, 2000.

[28] V. Moreau, Y. Renotte, and Y. Lion. Characterization of dupont photopolymer: determination of kinetic parameters in a diffusion model. *Applied Optics*, 41:3427–3435, 2002.

[29] V. Colvin, R. Larson, A. Harris, and M. Schilling. Quantitative model of volume hologram formation in photopolymers. *Journal of applied physics*, 81(9):5913–5923, 1997.

[30] L. Blaya, R. Carretero, and A. Mallavia. Holography as a technique for the study of photopolymerization kinetics in dry polymeric films with a nonlinear response. *Applied Optics*, 38:955–962, 1999.

[31] J. Kwon, H. Hwang, and K. Woo. Analysis of temporal behaviour of beams diffracted by volume gratings formed in photopolymers. *JOSA B*, 16:1651–1657, 1999.

[32] C. Neipp, S. Gallego, M. Ortuno, A. Márquez, M. L. Alvarez, A. Beléndez, and I. Pascual. First-harmonic diffusion-based model applied to a polyvinyl-alcohol–acrylamide-based photopolymer. *JOSA B*, 20(10):2052–2060, 2003.

[33] J. T. Sheridan and J. R. Lawrence. Nonlocal-response diffusion model of holographic recording in photopolymer. *JOSA A*, 17(6):1108–1114, 2000.

[34] J. Sheridan, M. Downey, and F. O'Neill. Diffusion based model of holographic grating formation in photopolymers: Generalised non-local material responses. *J. Opt. A: Pure and Applied Optics*, 3:477–488, 2001.

[35] I. Naydenova, R. Jallapuram, R. Howard, S. Martin, and V. Toal. Investigation of the diffusion processes in a self-processing acrylamide-based photopolymer system. *Applied Optics*, 43(14):2900–2905, 2004.

[36] D. Cody, I. Naydenova, and E. Mihaylova. Diacetone acrylamide-based non-toxic holographic photopolymer. In *Optical Modelling and Design II*, volume 8429, pages 404–410. SPIE, 2012.

[37] D. Cody, I. Naydenova, and E. Mihaylova. New non-toxic holographic photopolymer. *J. Opt.*, 14(1), 2012.

[38] D. Mackey, P. O'Reilly, and I. Naydenova. Theoretical modeling of the effect of polymer chain immobilization rates on holographic recording in photopolymers. *JOSA A*, 33(5):920–929, 2016.

[39] I. Aubrecht, M. Miler, and I. Koudela. Recording of holographic diffraction gratins in photopolymers: theoretical modeling and real-time monitoring of grating growth. *Journal of Modern Optics*, 45:1465–1477, 1998.

[40] R. Jallapuram, I. Naydenova, H. J. Byrne, S. Martin, R. Howard, and V. Toal. Raman spectroscopy for the characterization of the polymerization rate in an acrylamide-based photopolymer. *Applied Optics*, 47(2):206–212, 2008.

[41] P. W. Oliveira, H. Krug, P. Müller, and H. Schmidt. Fabrication of grin-materials by photopolymerization of diffusion-controlled organic-inorganic nanocomposite materials. *MRS Online Proceedings Library (OPL)*, 435, 1996.

[42] Y. Tomita, N. Suzuki, and K. Chikama. Holographic manipulation of nanoparticle distribution morphology in nanoparticle-dispersed photopolymers. *Optics Letters*, 30(8):839–841, 2005.

[43] N. Suzuki, Y. Tomita, and T. Kojima. Holographic recording in tio 2 nanoparticle-dispersed methacrylate photopolymer films. *Applied Physics Letters*, 81(22):4121–4123, 2002.

[44] C. Sanchez, M. Escuti, C. van Heesch, C. Bastiaansen, D. Broer, J. Loos, and R. Nussbaumer. Tio2 nanoparticle-photopolymer holographic recording. *Adv. Func. Mat.*, 15(10):1623–1629, 2005.

[45] Y. Tomita, K. Chikama, Y. Nohara, N. Suzuki, K. Furushima, and Y. Endoh. Two-dimensional imaging of atomic distribution morphology created by holographically induced mass transfer of monomer molecules and nanoparticles in a silica-nanoparticle-dispersed photopolymer film. *Optics Letters*, 31(10):1402–1404, 2006.

[46] N. Suzuki and Y. Tomita. Silica-nanoparticle-dispersed methacrylate photopolymers with net diffraction efficiency near 100%. *Applied Optics*, 43(10):2125–2129, 2004.

[47] L. Goldenberg, O. Sakhno, T. Smirnova, P. Helliwell, V. Chechik, and J. Stumpe. Holographic composites with gold nanoparticles: nanoparticles promote polymer segregation. *Chemistry of Materials*, 20:4619–4627, 2008.

[48] X. Xue, F. Hai, L. Gao, F. He, C. Li, Y. Li, and M. Huang. Effect of nanoparticle diameter on the holographic properties of gold nanoparticle dispersed acrylate photopolymer films. *Optik – Int. J. Light Electron Opt.*, 124(24):6987–6990, 2013.

[49] L. Balan, R. Schneider, and D. Lougnot. A new and convenient route to polyacrylate/silver nanocomposites by light-induced cross-linking polymerization. *Prog. Org. Coat.*, 62(3):351—357, 2008.

[50] L. Balan, C. Turck, O. Soppera, L. Vidal, and D. Lougnot. Holographic recording with polymer nanocomposites containing silver nanoparticles photogenerated in situ by the interference pattern. *Chemistry of Materials*, 21(24):5711—5718, 2009.

[51] L. Balan, J. Malval, R. Schneider, D. Le Nouen, and D. Lougnot. In-situ fabrication of polyacrylate–silver nanocomposite through photoinduced tandem reactions involving eosin dye. *Polymer*, 51(6):1363—1369, 2010.

[52] V. Pramitha, K. Nimmi, N. Subramanyan, R. Joseph, K. Sreekumar, and C. Kartha. Silver-doped photopolymer media for holographic recording. *Applied Optics*, 48(12):2255—2261, 2009.

[53] X. Liu, Y. Tomita, J. Oshima, K. Chikama, K. Matsubara, T. Nakashima, and T. Kawai. Holographic assembly of semiconductor cdse quantum dots in polymer for volume bragg grating structures with diffraction efficiency near 100%. *Appl. Phys. Lett.*, 95(26):2611091—2611093, 2009.

[54] G. Goourey, P. Claire, L. Balan, and Y. Israëli. Acrylate photopolymer doped with zno nanoparticles: an interesting candidate for photo-patterning applications. *J. Mat. Chem. C*, 1(21):3430—3438, 2013.

[55] D. Cody, E. Mihaylova, L. O'neill, T. Babeva, H. Awala, R. Retoux, S. Mintova, and I. Naydenova. Effect of zeolite nanoparticles on the optical properties of diacetone acrylamide-based photopolymer. *Optical Materials*, 37:181–187, 2014.

[56] E. Leite, I. Naydenova, S. Mintova, L. Leclercq, and V. Toal. Photopolymerizable nanocomposites for holographic recording and sensor application. *Applied Optics. Optical technology and biomedical optics*, 49(19):3652–3660, 2010.

[57] E. Leite. *Photopolymerizable nanocomposites for holographic applications.* PhD thesis, Technological University Dublin, 2010.

[58] T. Babeva, R. Todorov, and S. Mintova. Optical properties of silica mfi doped acrylamide-based photopolymer. *Journal of Optics A Pure and Applied Optics*, 11(2):024015, 2009.

[59] M. Moothanchery, I. Naydenova, S. Mintova, and V. Toal. Nanozeolites doped photopolymer layers with reduced shrinkage. *Optics Express*, 19(25):25786–25791, 2011.

[60] I. Naydenova, R. Jallaparam, and S. Martin. *Holographic humidity sensors*. Nova Science Publishers, 2011.

[61] I. Naydenova, R. Jallapuram, V. Toal, and S. Martin. A visual indication of environmental humidity using a color changing hologram recorded in a self-developing photopolymer. *Applied Physics Letters*, 92(3):031109, 2008.

[62] Izabela Naydenova, Svetlana Mintova, Suzanne Martin, and Vincent Toal. Nanocomposites for novel holographic applications. *SPIE News*, 2008.

[63] R. A. Vaia, C. L. Dennis, L. V. Natarajan, V. P. Tondiglia, D. W. Tomlin, and T. J. Bunning. One-step, micrometer-scale organization of nano-and mesoparticles using holographic photopolymerization: A generic technique. *Advanced Materials*, 13(20):1570–1574, 2001.

[64] Dana Mackey, Jack Lyons, and Izabela Naydenova. Modeling redistribution of nanozeolites in holographic recording. *Journal of the Optical Society of America A*, 41(7):1413–1419, 2024.

[65] Zhifu Xie. Cross-diffusion induced turing instability for a three species food chain model. *Journal of Mathematical Analysis and Applications*, 388(1):539–547, 2012.

[66] Yunfei Lv, Rong Yuan, and Yongzhen Pei. Turing pattern formation in a three species model with generalist predator and cross-diffusion. *Nonlinear Analysis: Theory, Methods & Applications*, 85:214–232, 2013.

[67] Baojun Miao. Persistence and turing instability in a cross-diffusive predator–prey system with generalist predator. *Advances in Difference Equations*, 2018:1–20, 2018.

[68] Masato Iida, Masayasu Mimura, and Hirokazu Ninomiya. Diffusion, cross-diffusion and competitive interaction. *Journal of mathematical biology*, 53(4):617–641, 2006.

[69] V. Toal. *Introduction to holography*. CRC press, 2022.

[70] H. Al Sunbul, N. Silikas, and D. C. Watts. Polymerization shrinkage kinetics and shrinkage-stress in dental resin-composites. *Dental Materials*, 32(8):998–1006, 2016.

[71] S. Gallego, A. Marquez, and D. Mendez. Analysis of pva/aa based photopolymers at the zero spatial frequency limit using interferometric methods. *Applied Optics*, 47(14):2557–2563, 2008.

[72] M. Moothanchery, I. Naydenova, and V. Toal. Study of the shrinkage caused by holographic grating formation in acrylamide based photopolymer film. *Optical Society of America*, 19(14):13395–13404, 2011.

[73] I. Naydenova and V. Toal. Nanoparticle doped photopolymers for holographic applications. In *Ordered porous solids*, pages 559–589. Elsevier, 2009.

[74] R. L. Sutherland, L. V. Natarajan, V. P. Tondiglia, T. J. Bunning, and W. W. Adams. Switchable holograms in new photopolymer-liquid crystal composite materials. In *Diffractive and Holographic Optics Technology II*, volume 2404, pages 132–143. International Society for Optics and Photonics, 1995.

[75] M. Moothanchery, S. Mintova, and I. Naydenova. Nanoparticle doped photopolymer with reduced shrinkage for holographic recording. In *Photonics 2010: International Conference on Fiber Optics and Photonics*, pages 11–15, Guwhatti, India, December 2010.

[76] Donald F Swinehart. The beer-lambert law. *Journal of chemical education*, 39(7):333, 1962.

[77] Suchuan Dong and Naxian Ni. A method for representing periodic functions and enforcing exactly periodic boundary conditions with deep neural networks. *Journal of Computational Physics*, 435:110242, 2021.

[78] J Caldwell and YY Kwan. Numerical methods for one-dimensional stefan problems. *Communications in numerical methods in engineering*, 20(7):535–545, 2004.

[79] M Vynnycky. On boundary immobilization for one-dimensional stefan-type problems with a moving boundary having initially parabolic-logarithmic behaviour. *Applied Mathematics and Computation*, 444:127803, 2023.

[80] Sean McGinty, Tuoi TN Vo, Martin Meere, Sean McKee, and Christopher McCormick. Some design considerations for polymer-free drug-eluting stents: a mathematical approach. *Acta biomaterialia*, 18:213–225, 2015.

[81] S. Mintova and T. Bein. Nanosized zeolite films for vapor-sensing applications. *Microporous and Mesoporous Materials*, 50(2-3):159–166, 2001.

[82] M. G. Moharam and T. K. Gaylord. Rigorous coupled-wave analysis of planar-grating diffraction. *JOSA*, 71(7):811–818, 1981.

[83] David A Edwards. A spatially nonlocal model for polymer-penetrant diffusion. *Zeitschrift für angewandte Mathematik und Physik ZAMP*, 52:254–288, 2001.

[84] Q. Liu and D. De Kee. Modelling of diffusion through nanocomposite membranes. *J. Non-Newtonian Fluid Mech.*, 131:32–43, 2005.

[85] J. L. M. Hurtado and C. R. Lowe. An integrated photonic-diffusion model for holographic sensors in polymeric matrices. *Journal of Membrane Science*, 495:14–19, 2015.

[86] S. Gul, S. Martin, J. Cassidy, and I. Naydenova. Development of sensitive holographic devices for physiological metal ion detection. *Proceedings of SPIE 10354*, 2017.

[87] J. Gallo and C. Verber. Model for the effects of material shrinkage on volume holograms. *Applied Optics*, 33:6797–6804, 1994.

Appendix A

Numerical Methods

Crank-Nicolson Scheme for General Reaction-Diffusion Equations

Reaction-diffusion systems are a common occurrence in nature, a chemical reaction that is non-uniform across a spatial dimension results in a concentration gradient which in turn drives mass transport. If $u = u(x, t)$ is the concentration of some mobile, reactive chemical with a diffusion coefficient D and $F(x, t; u)$ describes a photochemical reaction, the general form of a reaction-diffusion equation is

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} + F(x, t; u) \qquad 0 \leq x \leq 1; \quad t > 0. \tag{A.1}$$

This equation has the form of a parabolic 2nd order partial differential equation. The solution $u(x, t)$ defines a surface $\{x, t, u(x, t)\}$ over the domain of the (x, t)-plane. A numerical approximation of the solution can be generated by first discretizing the independent variables, $x \approx x_i$ and $t \approx t_j$ and defining constant finite differences

$$\Delta x = x_{i+1} - x_i, \qquad\qquad \Delta t = t_{j+1} - t_j.$$

The numerical solution is written as $u(x, t) \approx u(x_i, t_j) = u_i^j$. Assume the reaction function can be written as

$$F(x, t; u) = F_0 f(x) u,$$

where F_0 is a constant and f is some function that describes the spatial variation of the reaction; $F(x, t; u)$ is approximated as

$$F(x, t; u) \approx F_0 f(x_i) u_i^j = F_0 f_i u_i^j.$$

Eqn. A.1 is approximated by an Crank-Nicholson implicit finite difference scheme

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} = \frac{D}{2}\left[\frac{u_{i-1}^{j+1} - 2u_i^{j+1} + u_{i+1}^{j+1}}{\Delta x^2} + \frac{u_{i-1}^j - 2u_i^j + u_{i+1}^j}{\Delta x^2}\right] + F_0 f_i \left(\frac{u_i^j + u_i^{j+1}}{2}\right).$$

By defining $r = \Delta t / \Delta x^2$ the finite difference scheme can be rearranged as

$$-rDu_{i-1}^{j+1} + [2 + 2rD - F_0\Delta t f_i]u_i^{j+1} - rDu_{i+1}^{j+1} = rDu_{i-1}^{j}+ \tag{A.2}$$
$$[2 - 2rD + F_0\Delta t f_i]u_i^{j} + rDu_{i+1}^{j}.$$

Before the finite difference scheme can be solved the initial and boundary conditions must be addressed; a zero-flux boundary condition can be stated mathematically as

$$\left(\frac{\partial u}{\partial x}\right)_{x=0} = \left(\frac{\partial u}{\partial x}\right)_{x=1} = 0.$$

The condition can be approximated with a central finite difference scheme

$$u_{-1}^{j} = u_1^{j} \qquad\qquad u_{N-1}^{j} = u_{N+1}^{j}.$$

Eqn. A.2 can now be written in matrix form

$$\mathbf{A}\mathbf{u}^{j+1} = \mathbf{B}\mathbf{u}^{j},$$

$$\mathbf{u}^{j} = \begin{bmatrix} u_0^{j} & u_1^{j} & \dots & u_N^{j} \end{bmatrix}^{T},$$

$$\mathbf{A} = \begin{bmatrix} 2 + 2rD - F_0\Delta t f_0 & -2rD & \\ -rD & 2 + 2rD - F_0\Delta t f_1 & -rD \\ & & \ddots \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} 2 - 2rD + F_0\Delta t f_0 & 2rD & 0 \\ rD & 2 - 2rD + F_0\Delta t f_1 & rD \\ & & \ddots \end{bmatrix}.$$

If $\mathbf{A}$ is nonsingular then a numerical approximation of the solution at t_{j+1} is

$$\mathbf{u}^{j+1} = \mathbf{A}^{-1}\mathbf{B}\mathbf{u}^{j}. \tag{A.3}$$

Hence, if $u(x, t = 0) = \mathbf{u}^0$ is known then solving Eqn. A.3 gives $\mathbf{u}^1$, $\mathbf{u}^2$, $\mathbf{u}^3$, etc.

Eqn. 2.11

$$\frac{m_i^{k+1} - m_i^k}{\Delta t} = \frac{\alpha_{mm}}{2} \left[\frac{m_{i-1}^{k+1} - 2m_i^{k+1} + m_{i+1}^{k+1}}{\Delta x^2} + \frac{m_{i-1}^k - 2m_i^k + m_{i+1}^k}{\Delta x^2} \right] + \Phi^k \beta f_i^k \left(\frac{m_i^{k+1} + m_i^k}{2} \right),$$

$$\frac{p_i^{k+1} - p_i^k}{\Delta t} = \frac{\alpha_{pp}}{2} \left[\frac{p_{i-1}^{k+1} - 2p_i^{k+1} + p_{i+1}^{k+1}}{\Delta x^2} + \frac{p_{i-1}^k - 2p_i^k + p_{i+1}^k}{\Delta x^2} \right] +$$

$$\frac{\alpha_{pz}}{2} \left[\frac{z_{i-1}^k p_{i-1}^{k+1} - 2z_i^k p_i^{k+1} + z_{i+1}^k p_{i+1}^{k+1}}{\Delta x^2} + \frac{z_{i-1}^k p_{i-1}^k - 2z_i^k p_i^k + z_{i+1}^k p_{i+1}^k}{\Delta x^2} \right] +$$

$$\Phi^k \beta f_i^k \left(\frac{m_i^{k+1} + m_i^k}{2} \right) - \Phi^k \gamma p_i^k \left(\frac{p_i^k + p_i^{k+1}}{2} \right),$$

$$\frac{q_i^{k+1} - q_i^k}{\Delta t} = \Phi^k \gamma p_i^k \left(\frac{p_i^k + p_i^{k+1}}{2} \right),$$

$$\frac{z_i^{k+1} - z_i^k}{\Delta t} = \frac{\alpha_{zz}}{2} \left[\frac{z_{i-1}^{k+1} - 2z_i^{k+1} + z_{i+1}^{k+1}}{\Delta x^2} + \frac{z_{i-1}^k - 2z_i^k + z_{i+1}^k}{\Delta x^2} \right] +$$

$$\frac{\alpha_{zp}}{2} \left[\frac{p_{i-1}^k z_{i-1}^{k+1} - 2p_i^k z_i^{k+1} + p_{i+1}^k z_{i+1}^{k+1}}{\Delta x^2} + \frac{p_{i-1}^k z_{i-1}^k - 2p_i^k z_i^k + p_{i+1}^k z_{i+1}^k}{\Delta x^2} \right] +$$

$$\frac{\alpha_{zq}}{2} \left[\frac{q_{i-1}^k z_{i-1}^{k+1} - 2q_i^k z_i^{k+1} + q_{i+1}^k z_{i+1}^{k+1}}{\Delta x^2} + \frac{q_{i-1}^k z_{i-1}^k - 2q_i^k z_i^k + q_{i+1}^k z_{i+1}^k}{\Delta x^2} \right].$$

$$\left. \frac{\partial m}{\partial x} \right|_{x=0,1} = \left. \frac{\partial p}{\partial x} \right|_{x=0,1} = \left. \frac{\partial q}{\partial x} \right|_{x=0,1} = \left. \frac{\partial z}{\partial x} \right|_{x=0,1} = 0. \tag{A.4}$$

Eqn. A.4 can be discretized as

$$m_{-1}^k = m_{+1}^k \qquad p_{-1}^k = p_{+1}^k \qquad q_{-1}^k = q_{+1}^k \qquad z_{-1}^k = z_{+1}^k,$$
$$m_{N-1}^k = m_{N+1}^k \qquad p_{N-1}^k = p_{N+1}^k \qquad q_{N-1}^k = q_{N+1}^k \qquad z_{N-1}^k = z_{N+1}^k.$$

The numerical method was tested for $\Delta x = 0.01, 0.05, 0.1$ and 0.25; the results of which are presented in Fig. A.1.
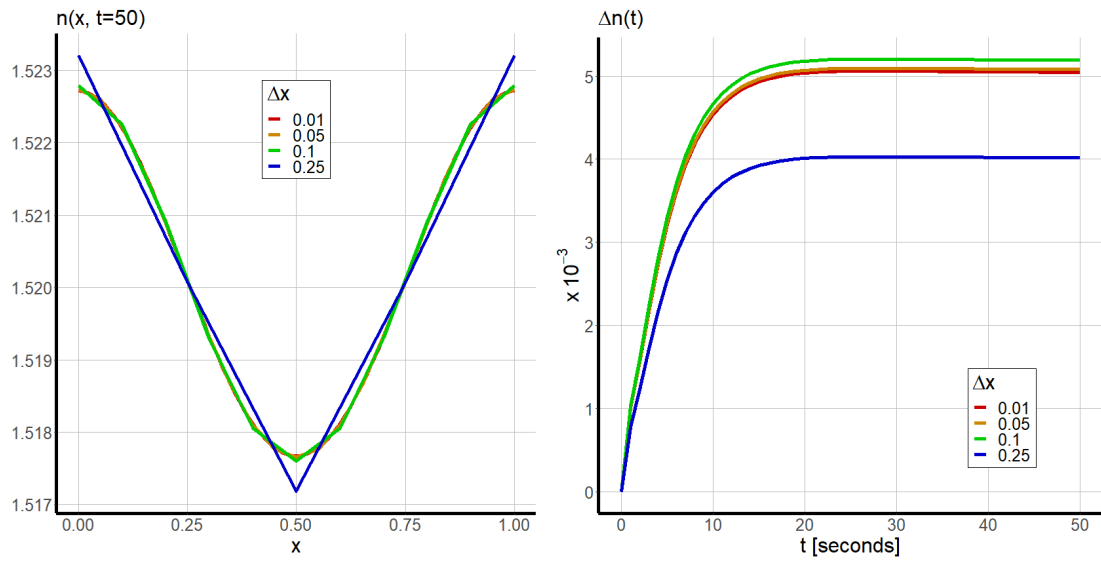
(a) Final RI profile   (b) RI modulation
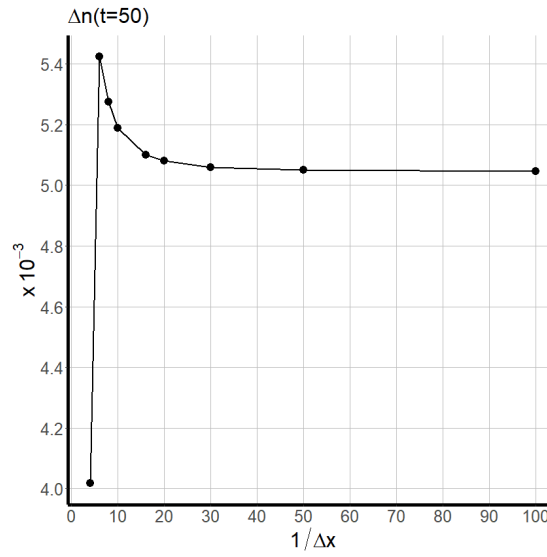
(c) Final RI modulation

Fig. A.1 Numerical results for different mesh sizes.

One can see, particularly in Fig. A.1c, the diminishing returns in the numerical results for $\Delta x \leq 0.05$.

Eqn. 3.14

If the nondimensionalized spatial variables are discretized

$$x \approx x_i, \qquad Y \approx Y_j, \qquad t \approx t_k \qquad i, j \in [0, 1, ..., J], \quad k \in [0, 1, 2, ...],$$

then for a numerical approximation

$$M(x, Y, t) \approx M_{i,j}^k, \quad P(x, Y, t) \approx P_{i,j}^k, \quad Q(x, Y, t) \approx Q_{i,j}^k, \quad Z(x, Y, t) \approx Z_{i,j}^k,$$
$$u(t) \approx u_k, \quad \phi_r(t) \approx \phi_r^k,$$

the boundary conditions are

$$M_{0,j}^k = M_{J,j}^k,$$
$$M_{1,j}^k - M_{-1,j}^k = M_{J+1,j}^k - M_{J-1,j}^k,$$
$$M_{1,j}^k - 2M_{0,j}^k + M_{-1,j}^k = M_{J+1,j}^k - 2M_{J,j}^k + M_{J-1,j}^k,$$
$$M_{i,-1}^k = M_{i,1}^k,$$
$$M_{i,J+1}^k = M_{i,J-1}^k,$$

$$P_{0,j}^k = P_{J,j}^k,$$
$$P_{1,j}^k - P_{-1,j}^k = P_{J+1,j}^k - P_{J-1,j}^k,$$
$$P_{1,j}^k - 2P_{0,j}^k + P_{-1,j}^k = P_{J+1,j}^k - 2P_{J,j}^k + P_{J-1,j}^k,$$
$$P_{i,-1}^k = P_{i,1}^k,$$
$$P_{i,J+1}^k = P_{i,J-1}^k,$$

$$Q_{i,-1}^k = Q_{i,1}^k,$$
$$Q_{i,J+1}^k = Q_{i,J-1}^k,$$

$$Z_{0,j}^k = Z_{J,j}^k,$$
$$Z_{1,j}^k - Z_{-1,j}^k = Z_{J+1,j}^k - Z_{J-1,j}^k,$$
$$Z_{1,j}^k - 2Z_{0,j}^k + Z_{-1,j}^k = Z_{J+1,j}^k - 2Z_{J,j}^k + Z_{J-1,j}^k,$$
$$Z_{i,-1}^k = Z_{i,1}^k,$$
$$Z_{i,J+1}^k = Z_{i,J-1}^k.$$

Initial conditions are

$$M_{i,j}^0 = 1, \quad P_{i,j}^0 = 0, \quad Q_{i,j}^0 = 0, \quad Z_{i,j}^0 = 1, \quad u_0 = 1.$$

The numerical scheme for Eqn. 3.14 is

$$\frac{M_{i,j}^{k+1} - M_{i,j}^k}{\Delta t} = \frac{Y_j}{u_k} \frac{u_k - u_{k-1}}{\Delta t} \left(\frac{M_{i,j+1}^{k+1} - M_{i,j-1}^{k+1}}{4\Delta Y} + \frac{M_{i,j+1}^k - M_{i,j-1}^k}{4\Delta Y} \right) +$$

$$\frac{\alpha_{mm}}{2} \left[\frac{M_{i-1,j}^{k+1} - 2M_{i,j}^{k+1} + M_{i+1,j}^{k+1}}{\Delta x^2} + \frac{M_{i-1,j}^k - 2M_{i,j}^k + M_{i+1,j}^k}{\Delta x^2} \right] +$$

$$\frac{\alpha_{mm}}{2u_k^2} \left[\frac{M_{i,j-1}^{k+1} - 2M_{i,j}^{k+1} + M_{i,j+1}^{k+1}}{\Delta Y^2} + \frac{M_{i,j-1}^k - 2M_{i,j}^k + M_{i,j+1}^k}{\Delta Y^2} \right] +$$

$$- \Phi^k \beta F_{i,j}^k \left(\frac{M_{i,j}^{k+1} + M_{i,j}^k}{2} \right),$$

$$\frac{P_{i,j}^{k+1} - P_{i,j}^k}{\Delta t} = \frac{Y_j}{u_k} \frac{u_k - u_{k-1}}{\Delta t} \left(\frac{P_{i,j+1}^{k+1} - M_{i,j-1}^{k+1}}{4\Delta Y} + \frac{P_{i,j+1}^k - M_{i,j-1}^k}{4\Delta Y} \right) +$$

$$\frac{\alpha_{pp}}{2} \left[\frac{P_{i-1,j}^{k+1} - 2P_{i,j}^{k+1} + P_{i+1,j}^{k+1}}{\Delta x^2} + \frac{P_{i-1,j}^k - 2P_{i,j}^k + P_{i+1,j}^k}{\Delta x^2} \right] +$$

$$\frac{\alpha_{pp}}{2u_k^2} \left[\frac{P_{i,j-1}^{k+1} - 2P_{i,j}^{k+1} + P_{i,j+1}^{k+1}}{\Delta Y^2} + \frac{P_{i,j-1}^k - 2P_{i,j}^k + P_{i,j+1}^k}{\Delta Y^2} \right] +$$

$$\frac{\alpha_{pz}}{2} \left[\frac{Z_{i-1,j}^k P_{i-1,j}^{k+1} - 2Z_{i,j}^k P_{i,j}^{k+1} + Z_{i+1,j}^k P_{i+1,j}^{k+1}}{\Delta x^2} + \right.$$

$$\left. \frac{Z_{i-1,j}^k P_{i-1,j}^k - 2Z_{i,j}^k P_{i,j}^k + Z_{i+1,j}^k P_{i+1,j}^k}{\Delta x^2} \right] +$$

$$\frac{\alpha_{pz}}{2u_k^2} \left[\frac{Z_{i,j-1}^k P_{i,j-1}^{k+1} - 2Z_{i,j}^k P_{i,j}^{k+1} + Z_{i,j+1}^k P_{i,j+1}^{k+1}}{\Delta Y^2} + \right.$$

$$\left. \frac{Z_{i,j-1}^k P_{i,j-1}^k - 2Z_{i,j}^k P_{i,j}^k + Z_{i,j+1}^k P_{i,j+1}^k}{\Delta Y^2} \right] +$$

$$\Phi^k \beta F_{i,j}^k \left(\frac{M_{i,j}^{k+1} + M_{i,j}^k}{2} \right) - \Phi^k \gamma P_{i,j}^k \left(\frac{P_{i,j}^k + P_{i,j}^{k+1}}{2} \right),$$

$$\frac{Q_{i,j}^{k+1} - Q_{i,j}^k}{\Delta t} = \frac{Y_j}{u_k} \frac{u_k - u_{k-1}}{\Delta t} \left(\frac{Q_{i,j+1}^{k+1} - Q_{i,j-1}^{k+1}}{4\Delta Y} + \frac{Q_{i,j+1}^k - Q_{i,j-1}^k}{4\Delta Y} \right) + \Phi^k \gamma P_{i,j}^k \left(\frac{P_{i,j}^k + P_{i,j}^{k+1}}{2} \right),$$

$$\frac{Z_{i,j}^{k+1} - Z_{i,j}^{k}}{\Delta t} = \frac{Y_j}{u_k} \frac{u_k - u_{k-1}}{\Delta t} \left(\frac{Z_{i,j+1}^{k+1} - Z_{i,j-1}^{k+1}}{4\Delta Y} + \frac{Z_{i,j+1}^{k} - Z_{i,j-1}^{k}}{4\Delta Y} \right) +$$

$$\frac{\alpha_{zz}}{2} \left[\frac{Z_{i-1,j}^{k+1} - 2Z_{i,j}^{k+1} + Z_{i+1,j}^{k+1}}{\Delta x^2} + \frac{Z_{i-1,j}^{k} - 2Z_{i,j}^{k} + Z_{i+1,j}^{k}}{\Delta x^2} \right] +$$

$$\frac{\alpha_{zz}}{2u_k^2} \left[\frac{Z_{i,j-1}^{k+1} - 2Z_{i,j}^{k+1} + Z_{i,j+1}^{k+1}}{\Delta Y^2} + \frac{Z_{i,j-1}^{k} - 2Z_{i,j}^{k} + Z_{i,j+1}^{k}}{\Delta Y^2} \right] +$$

$$\frac{\alpha_{zp}}{2} \left[\frac{P_{i-1,j}^{k} Z_{i-1,j}^{k+1} - 2P_{i,j}^{k} Z_{i,j}^{k+1} + P_{i+1,j}^{k} Z_{i+1,j}^{k+1}}{\Delta x^2} \right] +$$

$$\frac{\alpha_{zp}}{2} \left[\frac{P_{i-1,j}^{k} Z_{i-1,j}^{k} - 2P_{i,j}^{k} Z_{i,j}^{k} + P_{i+1,j}^{k} Z_{i+1,j}^{k}}{\Delta x^2} \right] +$$

$$\frac{\alpha_{zp}}{2u_k^2} \left[\frac{P_{i,j-1}^{k} Z_{i,j-1}^{k+1} - 2P_{i,j}^{k} Z_{i,j}^{k+1} + P_{i,j+1}^{k} Z_{i,j+1}^{k+1}}{\Delta Y^2} \right] +$$

$$\frac{\alpha_{zp}}{2u_k^2} \left[\frac{P_{i,j-1}^{k} Z_{i,j-1}^{k} - 2P_{i,j}^{k} Z_{i,j}^{k} + P_{i,j+1}^{k} Z_{i,j+1}^{k}}{\Delta Y^2} \right] +$$

$$\frac{\alpha_{zq}}{2} \left[\frac{Q_{i-1,j}^{k} Z_{i-1,j}^{k+1} - 2Q_{i,j}^{k} Z_{i,j}^{k+1} + Q_{i+1,j}^{k} Z_{i+1,j}^{k+1}}{\Delta x^2} \right] +$$

$$\frac{\alpha_{zq}}{2} \left[\frac{Q_{i-1,j}^{k} Z_{i-1,j}^{k} - 2Q_{i,j}^{k} Z_{i,j}^{k} + Q_{i+1,j}^{k} Z_{i+1,j}^{k}}{\Delta x^2} \right] +$$

$$\frac{\alpha_{zq}}{2u_k^2} \left[\frac{Q_{i,j-1}^{k} Z_{i,j-1}^{k+1} - 2Q_{i,j}^{k} Z_{i,j}^{k+1} + Q_{i,j+1}^{k} Z_{i,j+1}^{k+1}}{\Delta Y^2} \right] +$$

$$\frac{\alpha_{zq}}{2u_k^2} \left[\frac{Q_{i,j-1}^{k} Z_{i,j-1}^{k} - 2Q_{i,j}^{k} Z_{i,j}^{k} + Q_{i,j+1}^{k} Z_{i,j+1}^{k}}{\Delta Y^2} \right],$$

$$F_{i,j}^{k} = \exp\left[-a\zeta^* u_k (1 - Y_j) \right] \left[1 + \exp\left(-\xi^* Z_{i,j}^{k} \right) \cos\left(2\pi x - 2\pi \tan \phi_r^k u_k Y_j \right) \right].$$

$$\int_0^1 \int_0^1 M \, dx \, dY \approx M_k^* = \frac{\Delta x^2}{4} \Big[M_{0,0}^k + M_{0,J}^k + M_{J,0}^k + M_{J,J}^k + 2M_{0,1}^k + ... + 2M_{0,J-1}^k +$$

$$2M_{1,0}^k + ... + 2M_{J-1,0}^k + 2M_{1,J}^k + ... + 2M_{J-1,J}^k + 2M_{J,1}^k + ...$$

$$+ 2M_{J,J-1}^k + 4M_{2,2}^k + ... + 4M_{J-2,J-2}^k \Big],$$

$$\int_0^1 \int_0^1 P \, dx \, dY \approx P_k^* = \frac{\Delta x^2}{4} \Big[P_{0,0}^k + P_{0,J}^k + P_{J,0}^k + P_{J,J}^k + 2P_{0,1}^k + ... + 2P_{0,J-1}^k +$$

$$2P_{1,0}^k + ... + 2P_{J-1,0}^k + 2P_{1,J}^k + ... + 2P_{J-1,J}^k + 2P_{J,1}^k + ...$$

$$+ 2P_{J,J-1}^k + 4P_{2,2}^k + ... + 4P_{J-2,J-2}^k \Big],$$

$$\int_0^1 \int_0^1 Q \ dx \ dY \approx Q_k^* = \frac{\Delta x^2}{4} \Big[Q_{0,0}^k + Q_{0,J}^k + Q_{J,0}^k + Q_{J,J}^k + 2Q_{0,1}^k + ... + 2Q_{0,J-1}^k +$$
$$2Q_{1,0}^k + ... + 2Q_{J-1,0}^k + 2Q_{1,J}^k + ... + 2Q_{J-1,J}^k + 2Q_{J,1}^k + ...$$
$$+2Q_{J,J-1}^k + 4Q_{2,2}^k + ... + 4Q_{J-2,J-2}^k \Big],$$

$$u_k = \left[\frac{b_0}{\rho_b} + \frac{1}{\rho_m} + \frac{z_0}{\rho_z} \right]^{-1} \left[\frac{b_0}{\rho_b} + \frac{M_k^*}{\rho_m} + \frac{P_k^*}{\rho_p} + \frac{Q_k^*}{\rho_p} + \frac{z_0}{\rho_z} \right],$$
$$\phi_r^k = \tan^{-1} \left(\frac{\tan \phi_r^0}{u_k} \right),$$
$$\Lambda_k = \Lambda_0 \frac{\cos \phi_r^k}{\cos \phi_r^0}.$$

Eqn. 4.11

$$\frac{a_{i,j}^{k+1} - a_{i,j}^k}{\Delta t} = \alpha_{a,a} \left(\frac{a_{i+1,j}^{k+1} - 2a_{i,j}^{k+1} + a_{i-1,j}^{k+1}}{2\Delta x^2} + \frac{a_{i+1,j}^k - 2a_{i,j}^k + a_{i-1,j}^k}{2\Delta x^2} \right) +$$
$$\alpha_{a,a} \left(\frac{a_{i,j+1}^{k+1} - 2a_{i,j}^{k+1} + a_{i,j-1}^{k+1}}{2\Delta y^2} + \frac{a_{i,j+1}^k - 2a_{i,j}^k + a_{i,j-1}^k}{2\Delta y^2} \right) -$$
$$\gamma_{a,a} \left(\frac{a_{i,j}^{k+1} + a_{i,j}^k}{2} \right) ze_{i,j}^k + \omega_{a,a} za_{i,j}^k,$$
$$\frac{s_{i,j}^{k+1} - s_{i,j}^k}{\Delta t} = \alpha_{s,s} \left(\frac{s_{i+1,j}^{k+1} - 2s_{i,j}^{k+1} + s_{i-1,j}^{k+1}}{2\Delta x^2} + \frac{s_{i+1,j}^k - 2s_{i,j}^k + s_{i-1,j}^k}{2\Delta x^2} \right) +$$
$$\alpha_{s,s} \left(\frac{s_{i,j+1}^{k+1} - 2s_{i,j}^{k+1} + s_{i,j-1}^{k+1}}{2\Delta y^2} + \frac{s_{i,j+1}^k - 2s_{i,j}^k + s_{i,j-1}^k}{2\Delta y^2} \right) -$$
$$\gamma_{s,s} \left(\frac{s_{i,j}^{k+1} + s_{i,j}^k}{2} \right) ze_{i,j}^k + \omega_{s,s} zs_{i,j}^k,$$
$$\frac{ze_{i,j}^{k+1} - ze_{i,j}^k}{\Delta t} = \gamma_{a,z} a_{i,j}^k \left(\frac{ze_{i,j}^{k+1} + ze_{i,j}^k}{2} \right) + \omega_{a,z} za_{i,j}^k - \gamma_{s,z} s_{i,j}^k \left(\frac{ze_{i,j}^{k+1} + ze_{i,j}^k}{2} \right) + \omega_{s,z} zs_{i,j}^k,$$
$$\frac{za_{i,j}^{k+1} - za_{i,j}^k}{\Delta t} = \gamma_{a,z} a_{i,j}^k ze_{i,j}^k - \omega_{a,z} \left(\frac{za_{i,j}^{k+1} + za_{i,j}^k}{2} \right),$$
$$\frac{zs_{i,j}^{k+1} - zs_{i,j}^k}{\Delta t} = \gamma_{s,z} s_{i,j}^k ze_{i,j}^k - \omega_{s,z} \left(\frac{zs_{i,j}^{k+1} + zs_{i,j}^k}{2} \right).$$

Boundary conditions

$$a^k_{-1,j} = a^k_{J-1,j}, \qquad a^k_{J+1,j} = a^k_{1,j}, \qquad a^k_{i,-1} = a^k_{i,1}, \qquad a^k_{i,J+1} = a^k_{i,J-1},$$

$$s^k_{-1,j} = s^k_{J-1,j}, \qquad s^k_{J+1,j} = s^k_{1,j}, \qquad s^k_{i,-1} = s^k_{i,1}, \qquad s^k_{i,J+1} = s^k_{i,J-1},$$

$$ze^k_{-1,j} = ze^k_{J-1,j}, \qquad ze^k_{J+1,j} = ze^k_{1,j}, \qquad ze^k_{i,-1} = ze^k_{i,1}, \qquad ze^k_{i,J+1} = ze^k_{i,J-1},$$

$$za^k_{-1,j} = za^k_{J-1,j}, \qquad za^k_{J+1,j} = za^k_{1,j}, \qquad za^k_{i,-1} = za^k_{i,1}, \qquad za^k_{i,J+1} = za^k_{i,J-1},$$

$$zs^k_{-1,j} = zs^k_{J-1,j}, \qquad zs^k_{J+1,j} = zs^k_{1,j}, \qquad zs^k_{i,-1} = zs^k_{i,1}, \qquad zs^k_{i,J+1} = zs^k_{i,J-1}.$$

Appendix B

R Code: Functions

Unslanted Gratings

```r
coupled_cross_diffusion_v7 = function(

  end_exp=1e2, # End of exposure
  total.time=100,# Total simulation time
  lpmm=1e3,# Spatial frequency
  T0=50e-4,# cm
  I0=5,# Intensity of first recording beam
  xi=0.3,# Scattering coefficient
  Dm=1.6e-7,# Monomer diffusion coefficient
  nm=1.55,# Monomer refractive index
  rhom=1.15,# Monomer mass density
  Dp=6.35e-10,# Oligomer diffusivity ratio
  rhop=1.3,# Fractional van der Waals space loss
  np=1.56,# Oligomer refractive index
  nq=1.64,# Polymer refractive index
  Gamma=1,# Rate of immobilization
  Dz=1e-10,# Nanoparticle cross-diffusion ratio
  epsilon.mz=0,# Cross-diffusion
  epsilon.pz=13,# Cross-diffusion
  epsilon.qz=13,# Cross-diffusion
  wt.pc=5e-2,# Doping %
  rhoz=1.74,# Nanoparticle mass density
  nz=1.366,# Nanoparticle refractive index
  b0=5.05,# Binder mass
  nb=1.5,# Binder refractive inedx
  rhob=1.19, # Binder mass density
  lambda.probe=633e-7,# Wavelength of reconstruction beam
  Delta.t=1/100,# Numerical scheme time step
  Delta.x=1/20,# Numerical scheme spatial step
  output.time.step=1# Output time

){

  # 1.2 --- Define parameters
```

```r
35  iterations.per.second=1/Delta.t# Number of iterations each second
36  Nx=1/Delta.x + 1# Number of spatial points
37  x=seq(0, 1, length.out=Nx)# Non-dimensional grating distance
38  n_interations = total.time*iterations.per.second +1# Total number
      of iterations
39  r=Delta.t/Delta.x/Delta.x# Ratio of finite time step to squared
     finite spatial step
40  m0=1# Initial grams of monomer
41  t0=1 #  Reference time [s]
42  Lambda=1/10/lpmm # Grating period [cm]
43  j_start_exp=0/Delta.t # Iteration of exposure start
44  j_end_exp=end_exp/Delta.t # Iteration of exposure end
45  z0 = wt.pc/(1 - wt.pc)*(m0 + b0)# Initial nanoparticle mass
46  rep(1, Nx) -> m1# m at j=0
47  rep(0, Nx) -> p1# p at j=0
48  rep(0, Nx) -> q1# q at j=0
49  rep(1, Nx) -> z1# z at j=0
50
51  # 1.3 --- Non-dimensional parameters
52  alpha.m=Dm*t0/Lambda/Lambda# Monomer diffusion
53  F0=0.1*I0**0.3
54  beta=F0*t0# Monomer comsumption
55  alpha.p=Dp*t0/Lambda/Lambda# Oligomer diffusion
56  alpha.z=Dz*t0/Lambda/Lambda# Nondimensional self-nanoparticle
     diffusion
57  gamma = m0*Gamma*t0# Immobilization
58  alpha.mz=ifelse(wt.pc==0, 0, z0*epsilon.mz*alpha.m)
59  alpha.pz=ifelse(wt.pc==0, 0, z0*epsilon.pz*alpha.p)
60  alpha.zm=ifelse(wt.pc==0, 0, epsilon.mz*alpha.z)
61  alpha.zp=ifelse(wt.pc==0, 0, epsilon.pz*alpha.z)
62  alpha.zq=ifelse(wt.pc==0, 0, epsilon.qz*alpha.z)
63
64  matrix(m1, ncol=1) -> m# monomer density matrix
65  matrix(p1, ncol=1) -> p# oligomer density matrix
66  matrix(q1, ncol=1) -> q# Immobile polymer matrix
67  matrix(z1, ncol=1) -> z# Zeolite matrix
68
69  seq(0, total.time, by=output.time.step) -> time.vals
70  time.vals/Delta.t + 1 -> bind.iter
71  bind.iter = as.character(bind.iter[-1])
72
73  # 1.3 --- Calculate each time step via implicit finite difference
      method
74  for(j in 1:n_interations){
75
76    if(j >= j_start_exp & j <= j_end_exp){Phi=1} else {Phi = 0}#
     Phi=1 if illumination is on, 0 otherwise
77
78    f = 1 + exp(-xi*z0*z1)*cos(2*pi*x)# Illumination pattern
79    f = as.numeric(f)
```

```r
80
81    MM2 = diag(2 + Phi*Delta.t*beta*f, Nx)
82    MM1 = diag(2 - Phi*Delta.t*beta*f, Nx)
83    PP2 = diag(2 + Phi*Delta.t*gamma*as.numeric(p1), Nx)
84    PP1 = diag(2 - Phi*Delta.t*gamma*as.numeric(p1), Nx)
85    PM2 = diag(+Phi*Delta.t*beta*f, Nx)
86    PM1 = diag(+Phi*Delta.t*beta*f, Nx)
87    QQ2=diag(2, Nx)
88    QQ1=diag(2, Nx)
89    QP2=diag(+Phi*gamma*Delta.t*as.numeric(p1),Nx)
90    QP1=diag(+Phi*gamma*Delta.t*as.numeric(p1),Nx)
91    ZZ2 = diag(2, Nx)
92    ZZ1 = diag(2, Nx)
93
94    for(i in 1:(Nx)){
95
96      i.minus.1=ifelse(i == 1, i + 1, i - 1)
97      i.plus.1=ifelse(i == Nx, i - 1, i + 1)
98      i.minus.2=ifelse(i == 1, i + 2,ifelse(i == 2, i, i - 2))
99      i.plus.2=ifelse(i == Nx, i - 2,ifelse(i == Nx-1, i, i + 2))
100
101      MM2[i, i.minus.1] = MM2[i, i.minus.1] - r*alpha.m - r*alpha.
      mz*z1[i.minus.1]
102      MM2[i, i] = MM2[i, i] + 2*r*alpha.m + 2*r*alpha.mz*z1[i]
103      MM2[i, i.plus.1] = MM2[i, i.plus.1] - r*alpha.m - r*alpha.mz*
      z1[i.plus.1]
104
105      MM1[i, i.minus.1] = MM1[i, i.minus.1] + r*alpha.m + r*alpha.
      mz*z1[i.minus.1]
106      MM1[i, i] = MM1[i, i] - 2*r*alpha.m - 2*r*alpha.mz*z1[i]
107      MM1[i, i.plus.1] = MM1[i, i.plus.1] + r*alpha.m + r*alpha.mz*
      z1[i.plus.1]
108
109      PP2[i, i.minus.1] = PP2[i, i.minus.1] - r*alpha.p -  r*alpha.
      pz*z1[i.minus.1]
110      PP2[i, i] = PP2[i, i] + 2*r*alpha.p + 2*r*alpha.pz*z1[i]
111      PP2[i, i.plus.1] = PP2[i, i.plus.1] - r*alpha.p - r*alpha.pz*
      z1[i.plus.1]
112
113      PP1[i, i.minus.1] = PP1[i, i.minus.1] + r*alpha.p + r*alpha.
      pz*z1[i.minus.1]
114      PP1[i, i] = PP1[i, i] - 2*r*alpha.p - 2*r*alpha.pz*z1[i]
115      PP1[i, i.plus.1] = PP1[i, i.plus.1] + r*alpha.p + r*alpha.pz*
      z1[i.plus.1]
116
117      ZZ2[i, i.minus.1] = ZZ2[i, i.minus.1] - r*alpha.z - r*alpha.
      zq*q1[i.minus.1] - r*alpha.zp*p1[i.minus.1]  - r*alpha.zm*m1[i.
      minus.1]
118      ZZ2[i, i] = ZZ2[i, i] + 2*r*alpha.z + 2*r*alpha.zq*q1[i] + 2*
      r*alpha.zp*p1[i] + 2*r*alpha.zm*m1[i]
```

```r
119        ZZ2[i, i.plus.1] = ZZ2[i, i.plus.1] - r*alpha.z - r*alpha.zq*
      q1[i.plus.1]  - r*alpha.zp*p1[i.plus.1]  - r*alpha.zm*m1[i.plus
      .1]
120
121        ZZ1[i, i.minus.1] = ZZ1[i, i.minus.1] + r*alpha.z + r*alpha.
      zq*q1[i.minus.1] + r*alpha.zp*p1[i.minus.1] + r*alpha.zm*m1[i.
      minus.1]
122        ZZ1[i, i] = ZZ1[i, i] - 2*r*alpha.z - 2*r*alpha.zq*q1[i]  -
      2*r*alpha.zp*p1[i]  - 2*r*alpha.zm*m1[i]
123        ZZ1[i, i.plus.1] = ZZ1[i, i.plus.1] + r*alpha.z + r*alpha.zq*
      q1[i.plus.1] + r*alpha.zp*p1[i.plus.1] + r*alpha.zm*m1[i.plus.1]
124
125      }
126
127    m2 = solve(MM2) %*% MM1 %*% matrix(m1, ncol=1)
128
129    p2 = (solve(PP2) %*% PP1 %*% matrix(p1, ncol=1)) + (solve(PP2)
      %*% PM2 %*% matrix(m2, ncol=1)) + (solve(PP2) %*% PM1 %*% matrix
      (m1, ncol=1))
130
131    q2 = (solve(QQ2) %*% QQ1 %*% matrix(q1, ncol=1)) + (solve(QQ2)
      %*% QP2 %*% matrix(p2, ncol=1)) + (solve(QQ2) %*% QP1 %*% matrix
      (p1, ncol=1))
132
133    z2 = solve(ZZ2) %*% ZZ1 %*% matrix(z1, ncol=1)
134
135    if(as.character(j) %in% bind.iter){
136
137      matrix(cbind(m, m2), nrow=Nx) -> m
138      matrix(cbind(p, p2), nrow=Nx) -> p
139      matrix(cbind(q, q2), nrow=Nx) -> q
140      matrix(cbind(z, z2), nrow=Nx) -> z
141
142    }
143
144    m2 -> m1
145    p2 -> p1
146    q2 -> q1
147    z2 -> z1
148
149  }
150
151  # 2.1 --- Monomer density data frame
152  data.frame(x=seq(0,1,Delta.x),m) -> df.m# Turn the matrix into a
      data frame.
153  names(df.m)=c("x", paste0("t",time.vals[1:ncol(m)]))# Rename
      columns
154  df.m %>% melt(id.vars=c("x")) %>% rename(time=variable, m=value)
      %>% mutate(time=gsub(x=time, pattern="t", replacement=""), time=
```

```r
      as.numeric(time)) %>% arrange(time,x) -> df.m# Additional
      columns containing info on input

# 2.2 --- Oligomer density data frame
data.frame(x=seq(0,1,Delta.x), p) -> df.p# Turn the matrix into a
    data frame.
names(df.p)=c("x", paste0("t",time.vals[1:ncol(p)]))# Rename
  columns
df.p %>% melt(id.vars=c("x")) %>% rename(time=variable, p=value)
  %>% mutate(time=gsub(x=time, pattern="t", replacement=""), time=
  as.numeric(time)) %>% arrange(time,x) -> df.p# Additional
  columns containing info on input

# 2.3 --- Polymer density data frame
data.frame(x=seq(0,1,Delta.x), q) -> df.q# Turn the matrix into a
    data frame.
names(df.q)=c("x", paste0("t",time.vals[1:ncol(q)]))# Rename
  columns
df.q %>% melt(id.vars=c("x")) %>% rename(time=variable, q=value)
  %>% mutate(time=gsub(x=time, pattern="t", replacement=""), time=
  as.numeric(time)) %>% arrange(time,x) -> df.q# Additional
  columns containing info on input

# 2.4 --- Zeolite density data frame
data.frame(x=seq(0,1,Delta.x), z) -> df.z# Turn the matrix into a
    data frame.
names(df.z)=c("x", paste0("t",time.vals[1:ncol(z)]))# Rename
  columns
df.z %>% melt(id.vars=c("x")) %>% rename(time=variable, z=value)
  %>% mutate(time=gsub(x=time, pattern="t", replacement=""), time=
  as.numeric(time),z=z0*z) %>% arrange(time,x) -> df.z# Additional
    columns containing info on input

matrix(1, nrow=nrow(m), ncol=ncol(m)) -> b# Binder density matrix

Vb = b*b0/rhob # cm**3
Vm = m*m0/rhom # cm**3
Vp = p*m0/rhop # cm**3
Vq = q*m0/rhop # cm**3
Vz = z*z0/rhoz # cm**3
Vtotal=Vb+Vm+Vp+Vq+Vz # cm**3

phi.m = Vm/Vtotal
phi.b = Vb/Vtotal
phi.p = Vp/Vtotal
phi.q = Vq/Vtotal
phi.z = Vz/Vtotal
```

```r
186   Lorentz.Lorenz.RHS = phi.m*(nm*nm - 1)/(nm*nm + 2) + phi.b*(nb*nb
         - 1)/(nb*nb + 2) + phi.p*(np*np - 1)/(np*np + 2) + phi.q*(nq*nq
         - 1)/(nq*nq + 2) + phi.z*(nz*nz - 1)/(nz*nz + 2)
187   sqrt((2*Lorentz.Lorenz.RHS + 1)/(1 - Lorentz.Lorenz.RHS)) -> n
188
189   # 3.2 --- Refractive index data frame
190   data.frame(x=seq(0,1,Delta.x), n) -> df.n# Turn the matrix into a
         data frame.
191   names(df.n)=c("x", paste0("t",time.vals[1:ncol(n)]))# Rename
        columns
192   df.n %>% melt(id.vars=c("x")) %>% rename(time=variable, n=value)
        %>% mutate(time=gsub(x=time, pattern="t", replacement=""), time=
        as.numeric(time)) -> df.n
193
194   df = cbind(df.m,p=df.p$p,q=df.q$q,z=df.z$z,n=df.n$n)# %>% subset(
        time %in% time.vals)
195
196   # 4.1 --- Refractive index modulation data frame
197   Nt = ncol(n)
198   N0=rep(0, Nt)
199   N1=rep(0, Nt)
200   Q0=rep(0, Nt)
201   Q1=rep(0, Nt)
202   n.tilde=matrix(0, nrow=nrow(n), ncol=ncol(n))
203   diff=matrix(0, nrow=nrow(n), ncol=ncol(n))
204   n2=matrix(0, nrow=nrow(n), ncol=ncol(n))
205   d2=rep(0, Nt)
206   int.n2=rep(0,Nt)
207   h=matrix(0, nrow=Nx, ncol=Nt)
208   h0=m0/rhom + z0/rhoz + b0/rhob
209   shrinkage=rep(0,Nt)
210
211   for(k in 1:Nt){# For each time point ...
212
213     interior.points = 2:(Nx-1)
214     times.4 = interior.points[interior.points %% 2 == 0]
215     times.2 = interior.points[interior.points %% 2 != 0]
216
217     n0=n[, k]
218     n1=n[, k]*cos(2*pi*x)
219
220     N0[k]=Delta.x/3*(n0[1] + n0[Nx] + sum(2*n0[times.2]) + sum(4*n0
        [times.4]))
221     N1[k]=2*Delta.x/3*(n1[1] + n1[Nx] + sum(2*n1[times.2]) + sum(4*
        n1[times.4]))
222
223     q0=q[, k]
224     q1=q[, k]*cos(2*pi*x)
225
```

```r
226    Q0[k]=Delta.x/3*(q0[1] + q0[Nx] + sum(2*q0[times.2]) + sum(4*q0
       [times.4]))
227    Q1[k]=2*Delta.x/3*(q1[1] + q1[Nx] + sum(2*q1[times.2]) + sum(4*
       q1[times.4]))
228
229    n.tilde[, k] = N0[k] + N1[k]*cos(2*pi*x)
230    diff[, k] = (n[, k] - n.tilde[, k])**2
231    d2[k] = Delta.x/3*(diff[1, k] + diff[Nx, k] + sum(2*diff[times
       .2, k]) + sum(4*diff[times.4, k]))
232    n2[, k]=n[, k]**2
233    int.n2[k]=Delta.x/3*(n2[1, k] + sum(2*n2[times.2, k]) + sum(4*
       n2[times.4, k]) + n2[Nx, k])
234    h[,k]=(m[,k]/rhom + p[,k]/rhop + q[,k]/rhop + z[,k]*z0/rhoz +
       b0/rhob)/h0
235    int.h.dx=Delta.x/3*(h[1,k] + sum(4*h[times.4,k]) + sum(2*h[
       times.2,k]) + h[Nx,k])
236    shrinkage[k] = 1 - int.h.dx
237
238  }# End for loop
239
240  Delta.n=2*N1
241  distortion=d2/int.n2
242  time=time.vals[1:ncol(n)]# Time vector
243  redistributed=c()
244  for(i in time){
245    df %>% subset(time==i) %>% arrange(x) %>% mutate(v1=z/z0) %>%
       pull(v1) -> zi
246    ifelse(zi > 1, zi - 1, 0) -> zi.gt.1
247    Delta.x/3*(zi.gt.1[1] + sum(2*zi.gt.1[times.2]) + sum(4*zi.gt
       .1[times.4]) + zi.gt.1[Nx]) -> v
248    c(redistributed,v) -> redistributed
249  }
250  df.Dn=data.frame(time, shrinkage, distortion, Delta.n,
       redistributed, N0, Q1)
251  df=merge(df, df.Dn, by=c("time"))
252
253  df %>% mutate(end_exp=end_exp,total.time,lpmm,I0,xi,Dm,nm,rhom,
       rhop,Dp,np,nq,Gamma,wt.pc,Dz,epsilon.mz,epsilon.pz,epsilon.qz,
       rhoz,nz,nb,rhob,b0,z0,T0,T.t=T0*(1-shrinkage),alpha.m,beta,Delta
       .t,Delta.x,output.time.step,theta.B.t=asin(lambda.probe/2/N0/
       Lambda),nu=pi*Delta.n*T.t/lambda.probe/cos(theta.B.t),Model="
       coupled_cross_diffusion_v7") -> df
254
255  if(
256    nrow(
257      subset(
258        df,
259        z < 0
260      )
261    ) > 0
```

```
262    ){
263      return(cat("Fail: z(x,t) < 0"))
264    } else {
265      return(df)
266    }
267
268 }
```

Slanted Gratings

```
1 slanted_grating_simulation_v21 = function(
2
3   start_exp=0,# Start of exposure
4   end_exp=1e2, # End of exposure
5   total.time=100,# Total simulation time
6   lpmm=1e3,# Spatial frequency
7   I0=5,# Intensity of recording beam
8   slant.angle=1e-4,# Grating slant angle
9   xi=0.3,# Scattering coefficient
10  nm=1.55,# Monomer refractive index
11  rhom=1.15,# Monomer density
12  Dm=1.6e-7,# Monomer diffusion coefficient
13  Dp=6.35e-10,# Polymer diffusion coefficient
14  rhop=1.3,# Polymer density
15  np=1.56,# Oligomer refractive index
16  nq=1.64,# Polymer refractive index
17  Gamma=1,# Rate of immobilization
18  wt.pc=5e-2,# Doping %
19  Dz=1e-10,# Nanodopant self-diffusion coefficient
20  epsilon.pz=13,# Cross-diffusion ratio
21  epsilon.qz=13,# Cross-diffusion ratio
22  rhoz=1.74,# Nanodopant mass density
23  nz=1.366,# Nanodopant refractive index
24  b0=5.05,# Ratio of binder to monomer mass
25  nb=1.5,# Binder refractive index
26  rhob=1.19, # Binder mass density
27  T0=50e-4,# Depth of photosensitive layer [cm]
28  zeta=139,# absorption coefficient [cm**-1]
29  lambda.probe=633e-7,# Wavelength of reconstruction beam
30  Delta.t=1/100,# Numerical scheme time step
31  Delta.x=1/20,# Numerical scheme spatial step
32  output.time.step=1# Output time
33
34 ){
35
36   # 1.2 --- Define parameters
37   Delta.Y=Delta.x
38   iterations.per.second=1/Delta.t# Number of iterations each second
39   Nx=1/Delta.x + 1# Number of spatial points
```

```
40  Ny=1/Delta.Y + 1# Number of spatial points
41  if(Nx%%2==0){return(warning("Number of x mesh points must be an
       odd number."))}
42  if(Ny%%2==0){return(warning("Number of y mesh points must be an
       odd number."))}
43  x=seq(0, 1, length.out=Nx)# Non-dimensional grating distance
44  y=seq(0,1,length.out=Ny)# Non-dimensional depth
45  Nt = total.time*iterations.per.second+1# Total number of
       iterations
46  r=Delta.t/Delta.x/Delta.x# Ratio of finite time step to squared
       finite spatial step
47  m0=1# Initial mass of monomer
48  t0=1 #  Reference time [s]
49  Lambda0=1/10/lpmm # Grating period [cm]
50  Lambda1=Lambda0
51  Lambda.t=c(Lambda0)
52  j_start_exp=start_exp/Delta.t # Iteration of exposure start
53  j_end_exp=end_exp/Delta.t # Iteration of exposure end
54  z0 = wt.pc/(1 - wt.pc)*(m0 + b0)# Initial nanodopant to monomer
55
56  # 1.3 --- Matrix initial conditions
57  u1=1
58  u.t=c(u1)
59  du.dt=0
60  rep(1, Nx*Nx) -> m1# m at j=0
61  rep(0, Nx*Nx) -> p1# p at j=0
62  rep(0, Nx*Nx) -> q1# q at j=0
63  rep(1, Nx*Nx) -> z1# z at j=0
64  rep(1, Nx*Nx) -> b1# b at j=0
65
66  matrix(m1, ncol=1) -> m# monomer density matrix
67  matrix(p1, ncol=1) -> p# oligomer density matrix
68  matrix(q1, ncol=1) -> q# Immobile polymer matrix
69  matrix(z1, ncol=1) -> z# Zeolite matrix
70  matrix(b1, ncol=1) -> b# Binder density matrix
71
72  Volume0=m0/rhom + b0/rhob + z0/rhoz
73  Volume.t=c(Volume0)
74
75  phi.m0=m0/rhom/Volume0
76  phi.z0=z0/rhoz/Volume0
77  phi.b0=b0/rhob/Volume0
78
79  Lorentz.Lorenz.RHS = phi.m0*(nm*nm - 1)/(nm*nm + 2) + phi.b0*(nb*
       nb - 1)/(nb*nb + 2) + phi.z0*(nz*nz - 1)/(nz*nz + 2)
80  Initial.RI=sqrt((2*Lorentz.Lorenz.RHS + 1)/(1 - Lorentz.Lorenz.
       RHS))
81  slant.angle=ifelse(slant.angle==0,1e-5,slant.angle)
82  phi.0=slant.angle/180*pi
83  phi.r0=asin(sin(phi.0)/Initial.RI)
```

```r
84    phi.r1=phi.r0
85    phi.r.t=c(phi.r0)
86    theta_B0=asin(lambda.probe/2/Initial.RI/Lambda0) - phi.r0
87    y.hat0=Lambda0/sin(phi.r0)
88    y.hat1=y.hat0
89    y.hat.t=c(y.hat0)
90    x.hat=Lambda0/cos(phi.r0)
91
92    # 1.4 --- time step iterations
93    seq(0, total.time, by=output.time.step) -> time.vals
94    time.vals/Delta.t + 1 -> bind.iter
95    bind.iter = as.character(bind.iter[-1])
96
97    # 1.5 --- Nondimensionalized parameters
98    alpha.m.x=Dm*t0/x.hat/x.hat
99    alpha.m.y=Dm*t0/T0/T0
100   alpha.p.x=Dp*t0/x.hat/x.hat
101   alpha.p.y=Dp*t0/T0/T0
102   alpha.z.x=Dz*t0/x.hat/x.hat
103   alpha.z.y=Dz*t0/T0/T0
104   F0=0.1*I0**0.3
105   beta=F0*t0
106   gamma=Gamma*m0*t0
107   zeta_star=zeta*T0
108   T_star=T0/x.hat
109   xi_star=xi*z0
110   #beta=matrix(0, nrow=Nx, ncol=Nx)
111   #for(k in 1:Nx){beta[,k]=F0*t0*exp(-zeta_star*Y[k])}
112   #beta=melt(beta)$value
113   #gamma=Gamma*t0
114
115   interior.points = 2:(Nx-1)
116   times.4 = interior.points[interior.points %% 2 == 0]
117   times.2 = interior.points[interior.points %% 2 != 0]
118   x=seq(0, 1, Delta.x)
119   Y=seq(0,1, Delta.x)
120   time=seq(0,total.time,output.time.step)
121   Y1=sort(rep(Y, Nx))
122   int_M=c(1)
123   int_P=c(0)
124   int_Q=c(0)
125   int_Z=c(z0)
126   int_B=c(b0)
127   Actual.Shrinkage=c(0)
128
129   # 1.6 --- Calculate each time step via implicit finite difference
          method
130   for(j in 1:Nt){
131
```

```r
132    if(j >= j_start_exp & j <= j_end_exp){Phi=1} else {Phi = 0}#
      Phi=1 if illumination is on, 0 otherwise

133

134    f = matrix(0, nrow=Nx, ncol=Nx)
135    #gamma=matrix(0, nrow=Nx, ncol=Nx)
136    for(i in 1:Nx){
137       matrix.z1=matrix(z1, nrow=Nx, ncol=Nx)
138       z1.i=as.numeric(matrix.z1[,i])
139       f[,i] = exp(-0.3*zeta_star*u1*(1-Y[i]))*(1 + exp(-xi_star*z1.
      i)*cos(2*pi*x - 2*pi*T0/y.hat1*u1*Y[i]))
140    }
141    f=melt(f)$value

142

143    MM2 = diag(2 + Phi*Delta.t*beta*f, Nx*Nx)
144    MM1 = diag(2 - Phi*Delta.t*beta*f, Nx*Nx)
145    PP2 = diag(2 + Delta.t*gamma*as.numeric(p1), Nx*Nx)
146    PP1 = diag(2 - Delta.t*gamma*as.numeric(p1), Nx*Nx)
147    PM2 = diag(+Phi*Delta.t*beta*f, Nx*Nx)
148    PM1 = diag(+Phi*Delta.t*beta*f, Nx*Nx)
149    QQ2=diag(2, Nx*Nx)
150    QQ1=diag(2, Nx*Nx)
151    QP2=diag(+Phi*gamma*Delta.t*as.numeric(p1),Nx*Nx)
152    QP1=diag(+Phi*gamma*Delta.t*as.numeric(p1),Nx*Nx)
153    ZZ2 = diag(2, Nx*Nx)
154    ZZ1 = diag(2, Nx*Nx)
155    BB2 = diag(2, Nx*Nx)
156    BB1 = diag(2, Nx*Nx)

157

158    for(i in 1:(Nx*Nx)){

159

160       i.minus.1=ifelse((i + Nx)%%Nx == 1, i + Nx - 2, i - 1)

161

162       i.plus.1=ifelse((i + Nx)%%Nx == 0, i - Nx + 2, i + 1)

163

164       j.minus.1=ifelse(i < Nx+1, i + Nx, i - Nx)

165

166       j.plus.1=ifelse(i > (Nx-1)*Nx, i - Nx, i + Nx)

167

168       # i.minus.2=ifelse((i + Nx)%%Nx == 1, i + Nx - 3, ifelse((i +
       Nx)%%Nx == 2, i + Nx - 2, i - 2))
169       #
170       # i.plus.2=ifelse((i + Nx)%%Nx == 0, i - Nx + 3, ifelse((i +
      Nx)%%Nx == Nx-1, i - Nx + 2, i + 2))
171       #
172       # j.minus.2=ifelse(i <= Nx, i + 2*Nx, ifelse(i <= 2*Nx, i, i
       - 2*Nx))
173       #
174       # j.plus.2=ifelse(i > (Nx-1)*Nx, i - 2*Nx, ifelse(i > (Nx-2)*
      Nx, i, i + 2*Nx))

175
```

```
176      MM2[i, i.minus.1] = MM2[i, i.minus.1] - r*alpha.m.x
177      MM2[i, j.minus.1] = MM2[i, j.minus.1] - r*alpha.m.y/u1/u1
178      MM2[i, i] = MM2[i, i] + 2*r*alpha.m.x
179      MM2[i, i] = MM2[i, i] + 2*r*alpha.m.y/u1/u1
180      MM2[i, i.plus.1] = MM2[i, i.plus.1] - r*alpha.m.x
181      MM2[i, j.plus.1] = MM2[i, j.plus.1] - r*alpha.m.y/u1/u1
182
183      MM1[i, i.minus.1] = MM1[i, i.minus.1] + r*alpha.m.x
184      MM1[i, j.minus.1] = MM1[i, j.minus.1] + r*alpha.m.y/u1/u1
185      MM1[i, i] = MM1[i, i] - 2*r*alpha.m.x
186      MM1[i, i] = MM1[i, i] - 2*r*alpha.m.y/u1/u1
187      MM1[i, i.plus.1] = MM1[i, i.plus.1] + r*alpha.m.x
188      MM1[i, j.plus.1] = MM1[i, j.plus.1] + r*alpha.m.y/u1/u1
189
190      PP2[i, i.minus.1] = PP2[i, i.minus.1] - r*alpha.p.x*(1 +
     epsilon.pz*z0*z1[i.minus.1])
191      PP2[i, j.minus.1] = PP2[i, j.minus.1] - r*alpha.p.y/u1/u1*(1
     + epsilon.pz*z0*z1[j.minus.1])
192      PP2[i, i] = PP2[i, i] + 2*r*alpha.p.x*(1 + epsilon.pz*z0*z1[i
     ])
193      PP2[i, i] = PP2[i, i] + 2*r*alpha.p.y/u1/u1*(1 + epsilon.pz*
     z0*z1[i])
194      PP2[i, i.plus.1] = PP2[i, i.plus.1] - r*alpha.p.x*(1 +
     epsilon.pz*z0*z1[i.plus.1])
195      PP2[i, j.plus.1] = PP2[i, j.plus.1] - r*alpha.p.y/u1/u1*(1 +
     epsilon.pz*z0*z1[j.plus.1])
196
197      PP1[i, i.minus.1] = PP1[i, i.minus.1] + r*alpha.p.x*(1 +
     epsilon.pz*z0*z1[i.minus.1])
198      PP1[i, j.minus.1] = PP1[i, j.minus.1] + r*alpha.p.y/u1/u1*(1
     + epsilon.pz*z0*z1[j.minus.1])
199      PP1[i, i] = PP1[i, i] - 2*r*alpha.p.x*(1 + epsilon.pz*z0*z1[i
     ])
200      PP1[i, i] = PP1[i, i] - 2*r*alpha.p.y/u1/u1*(1 + epsilon.pz*
     z0*z1[i])
201      PP1[i, i.plus.1] = PP1[i, i.plus.1] + r*alpha.p.x*(1 +
     epsilon.pz*z0*z1[i.plus.1])
202      PP1[i, j.plus.1] = PP1[i, j.plus.1] + r*alpha.p.y/u1/u1*(1 +
     epsilon.pz*z0*z1[j.plus.1])
203
204      ZZ2[i, i.minus.1] = ZZ2[i, i.minus.1] - r*alpha.z.x*(1 +
     epsilon.qz*q1[i.minus.1] + epsilon.pz*p1[i.minus.1])
205      ZZ2[i, j.minus.1] = ZZ2[i, j.minus.1] - r*alpha.z.y/u1/u1*(1
     + epsilon.qz*q1[j.minus.1] + epsilon.pz*p1[j.minus.1])
206      ZZ2[i, i] = ZZ2[i, i] + 2*r*alpha.z.x*(1 + epsilon.qz*q1[i] +
      epsilon.pz*p1[i])
207      ZZ2[i, i] = ZZ2[i, i] + 2*r*alpha.z.y/u1/u1*(1 + epsilon.qz*
     q1[i] + epsilon.pz*p1[i])
208      ZZ2[i, i.plus.1] = ZZ2[i, i.plus.1] - r*alpha.z.x*(1 +
     epsilon.qz*q1[i.plus.1] + epsilon.pz*p1[i.plus.1])
```

```
209     ZZ2[i, j.plus.1] = ZZ2[i, j.plus.1] - r*alpha.z.y/u1/u1*(1 +
        epsilon.qz*q1[j.plus.1] + epsilon.pz*p1[j.plus.1])
210
211     ZZ1[i, i.minus.1] = ZZ1[i, i.minus.1] + r*alpha.z.x*(1 +
        epsilon.qz*q1[i.minus.1] + epsilon.pz*p1[i.minus.1])
212     ZZ1[i, j.minus.1] = ZZ1[i, j.minus.1] + r*alpha.z.y/u1/u1*(1
        + epsilon.qz*q1[j.minus.1] + epsilon.pz*p1[j.minus.1])
213     ZZ1[i, i] = ZZ1[i, i] - 2*r*alpha.z.x*(1 + epsilon.qz*q1[i] +
         epsilon.pz*p1[i])
214     ZZ1[i, i] = ZZ1[i, i] - 2*r*alpha.z.y/u1/u1*(1 + epsilon.qz*
        q1[i] + epsilon.pz*p1[i])
215     ZZ1[i, i.plus.1] = ZZ1[i, i.plus.1] + r*alpha.z.x*(1 +
        epsilon.qz*q1[i.plus.1] + epsilon.pz*p1[i.plus.1])
216     ZZ1[i, j.plus.1] = ZZ1[i, j.plus.1] + r*alpha.z.y/u1/u1*(1 +
        epsilon.qz*q1[j.plus.1] + epsilon.pz*p1[j.plus.1])
217
218     MM2[i, j.minus.1] = MM2[i, j.minus.1] + Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
219     MM1[i, j.minus.1] = MM1[i, j.minus.1] - Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
220     MM2[i, j.plus.1] = MM2[i, j.plus.1] - Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
221     MM1[i, j.plus.1] = MM1[i, j.plus.1] + Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
222
223     PP2[i, j.minus.1] = PP2[i, j.minus.1] + Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
224     PP1[i, j.minus.1] = PP1[i, j.minus.1] - Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
225     PP2[i, j.plus.1] = PP2[i, j.plus.1] - Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
226     PP1[i, j.plus.1] = PP1[i, j.plus.1] + Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
227
228     QQ2[i, j.minus.1] = QQ2[i, j.minus.1] + Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
229     QQ1[i, j.minus.1] = QQ1[i, j.minus.1] - Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
230     QQ2[i, j.plus.1] = QQ2[i, j.plus.1] - Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
231     QQ1[i, j.plus.1] = QQ1[i, j.plus.1] + Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
232
233     ZZ2[i, j.minus.1] = ZZ2[i, j.minus.1] + Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
234     ZZ1[i, j.minus.1] = ZZ1[i, j.minus.1] - Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
235     ZZ2[i, j.plus.1] = ZZ2[i, j.plus.1] - Y1[i]/u1*Delta.t/2/
        Delta.x*du.dt
```

```r
236      ZZ1[i, j.plus.1] = ZZ1[i, j.plus.1] + Y1[i]/u1*Delta.t/2/
     Delta.x*du.dt

237

238      BB2[i, j.minus.1] = BB2[i, j.minus.1] + Y1[i]/u1*Delta.t/2/
     Delta.x*du.dt
239      BB1[i, j.minus.1] = BB1[i, j.minus.1] - Y1[i]/u1*Delta.t/2/
     Delta.x*du.dt
240      BB2[i, j.plus.1] = BB2[i, j.plus.1] - Y1[i]/u1*Delta.t/2/
     Delta.x*du.dt
241      BB1[i, j.plus.1] = BB1[i, j.plus.1] + Y1[i]/u1*Delta.t/2/
     Delta.x*du.dt

242

243

244    }

245

246   m2 = solve(MM2) %*% MM1 %*% matrix(m1, ncol=1)

247

248   p2 = solve(PP2) %*% ((PP1 %*% matrix(p1, ncol=1)) + (PM2 %*%
     matrix(m2, ncol=1)) + (PM1 %*% matrix(m1, ncol=1)))

249

250   q2 = solve(QQ2) %*% ((QQ1 %*% matrix(q1, ncol=1)) + (QP2 %*%
     matrix(p2, ncol=1)) + (QP1 %*% matrix(p1, ncol=1)))

251

252   if(z0==0){
253     z2 = matrix(z1, ncol=1)
254   } else {
255     z2 = solve(ZZ2) %*% ZZ1 %*% matrix(z1, ncol=1)
256   }

257

258   b2 = solve(BB2) %*% BB1 %*% matrix(b1, ncol=1)

259

260

261   dm.dt = matrix((m2-m1)/Delta.t, ncol=Nx)
262   dp.dt = matrix((p2-p1)/Delta.t, ncol=Nx)
263   dq.dt = matrix((q2-q1)/Delta.t, ncol=Nx)
264   dz.dt = matrix(z0*(z2-z1)/Delta.t, ncol=Nx)

265

266   int_dmdt=Delta.x*Delta.x/4*(dm.dt[1,1] + dm.dt[Nx,1] + dm.dt[1,
     Nx] + dm.dt[Nx,Nx] + sum(2*dm.dt[1,(2:(Nx-1))]) + sum(2*dm.dt
     [(2:(Nx-1)),1]) + sum(2*dm.dt[Nx,(2:(Nx-1))]) + sum(2*dm.dt[(2:(
     Nx-1)),Nx]) + sum(4*dm.dt[(2:(Nx-1)),(2:(Nx-1))]) )

267

268   int_dpdt=Delta.x*Delta.x/4*(dp.dt[1,1] + dp.dt[Nx,1] + dp.dt[1,
     Nx] + dp.dt[Nx,Nx] + sum(2*dp.dt[1,(2:(Nx-1))]) + sum(2*dp.dt
     [(2:(Nx-1)),1]) + sum(2*dp.dt[Nx,(2:(Nx-1))]) + sum(2*dp.dt[(2:(
     Nx-1)),Nx]) + sum(4*dp.dt[(2:(Nx-1)),(2:(Nx-1))]) )

269

270   int_dqdt=Delta.x*Delta.x/4*(dq.dt[1,1] + dq.dt[Nx,1] + dq.dt[1,
     Nx] + dq.dt[Nx,Nx] + sum(2*dq.dt[1,(2:(Nx-1))]) + sum(2*dq.dt
```

```
      [(2:(Nx-1)),1]) + sum(2*dq.dt[Nx,(2:(Nx-1))]) + sum(2*dq.dt[(2:(
      Nx-1)),Nx]) + sum(4*dq.dt[(2:(Nx-1)),(2:(Nx-1))]) )
271
272   int_dzdt=Delta.x*Delta.x/4*(dz.dt[1,1] + dz.dt[Nx,1] + dz.dt[1,
      Nx] + dz.dt[Nx,Nx] + sum(2*dz.dt[1,(2:(Nx-1))]) + sum(2*dz.dt
      [(2:(Nx-1)),1]) + sum(2*dz.dt[Nx,(2:(Nx-1))]) + sum(2*dz.dt[(2:(
      Nx-1)),Nx]) + sum(4*dz.dt[(2:(Nx-1)),(2:(Nx-1))]) )
273
274   m2 -> m1
275   p2 -> p1
276   q2 -> q1
277   z2 -> z1
278   b2 -> b1
279
280   m.t=matrix(m0*m1, ncol=Nx)
281   p.t=matrix(m0*p1, ncol=Nx)
282   q.t=matrix(m0*q1, ncol=Nx)
283   z.t=matrix(z0*z1, ncol=Nx)
284   b.t=matrix(b0*b1, ncol=Nx)
285
286   int_M1=Delta.x*Delta.x/4*(m.t[1,1] + m.t[Nx,1] + m.t[1,Nx] + m.
      t[Nx,Nx] + sum(2*m.t[1,(2:(Nx-1))]) + sum(2*m.t[(2:(Nx-1)),1]) +
       sum(2*m.t[Nx,(2:(Nx-1))]) + sum(2*m.t[(2:(Nx-1)),Nx]) + sum(4*m
      .t[(2:(Nx-1)),(2:(Nx-1))]) )
287
288   int_P1=Delta.x*Delta.x/4*(p.t[1,1] + p.t[Nx,1] + p.t[1,Nx] + p.
      t[Nx,Nx] + sum(2*p.t[1,(2:(Nx-1))]) + sum(2*p.t[(2:(Nx-1)),1]) +
       sum(2*p.t[Nx,(2:(Nx-1))]) + sum(2*p.t[(2:(Nx-1)),Nx]) + sum(4*p
      .t[(2:(Nx-1)),(2:(Nx-1))]) )
289
290   int_Q1=Delta.x*Delta.x/4*(q.t[1,1] + q.t[Nx,1] + q.t[1,Nx] + q.
      t[Nx,Nx] + sum(2*q.t[1,(2:(Nx-1))]) + sum(2*q.t[(2:(Nx-1)),1]) +
       sum(2*q.t[Nx,(2:(Nx-1))]) + sum(2*q.t[(2:(Nx-1)),Nx]) + sum(4*q
      .t[(2:(Nx-1)),(2:(Nx-1))]) )
291
292   int_Z1=Delta.x*Delta.x/4*(z.t[1,1] + z.t[Nx,1] + z.t[1,Nx] + z.
      t[Nx,Nx] + sum(2*z.t[1,(2:(Nx-1))]) + sum(2*z.t[(2:(Nx-1)),1]) +
       sum(2*z.t[Nx,(2:(Nx-1))]) + sum(2*z.t[(2:(Nx-1)),Nx]) + sum(4*z
      .t[(2:(Nx-1)),(2:(Nx-1))]) )
293
294   int_B1=Delta.x*Delta.x/4*(b.t[1,1] + b.t[Nx,1] + b.t[1,Nx] + b.
      t[Nx,Nx] + sum(2*b.t[1,(2:(Nx-1))]) + sum(2*b.t[(2:(Nx-1)),1]) +
       sum(2*b.t[Nx,(2:(Nx-1))]) + sum(2*b.t[(2:(Nx-1)),Nx]) + sum(4*b
      .t[(2:(Nx-1)),(2:(Nx-1))]) )
295
296   du.dt = (int_dmdt/rhom + int_dpdt/rhop + int_dqdt/rhop + int_
      dzdt/rhoz)/Volume0
297
298   u2 = u1 + Delta.t*du.dt; u2 -> u1
299
```

```r
300      phi.r1=atan(tan(phi.r0)/u1)
301
302      Lambda1=cos(phi.r1)/cos(phi.r0)*Lambda0
303
304      y.hat1=Lambda1/sin(phi.r1)
305
306      if(as.character(j) %in% bind.iter){
307
308        matrix(cbind(m, m1), nrow=Nx*Nx) -> m
309        matrix(cbind(p, p1), nrow=Nx*Nx) -> p
310        matrix(cbind(q, q1), nrow=Nx*Nx) -> q
311        matrix(cbind(z, z1), nrow=Nx*Nx) -> z
312        matrix(cbind(b, b1), nrow=Nx*Nx) -> b
313        Actual.Shrinkage=c(Actual.Shrinkage,1-u1)
314        u.t=c(u.t,u1)
315        Lambda.t=c(Lambda.t,Lambda1)
316        phi.r.t=c(phi.r.t,phi.r1)
317        y.hat.t=c(y.hat.t,y.hat1)
318        int_M=c(int_M,int_M1)
319        int_P=c(int_P,int_P1)
320        int_Q=c(int_Q,int_Q1)
321        int_Z=c(int_Z,int_Z1)
322        int_B=c(int_B,int_B1)
323
324      }
325
326
327    }
328
329    # 2.1 --- Monomer density data frame
330    data.frame(x=rep(seq(0,1,Delta.x), Nx), Y=sort(rep(seq(0,1,Delta.
       x), Nx)), m) -> df.m# Turn the matrix into a data frame.
331    names(df.m)=c("x","Y", paste0("t",time.vals[1:ncol(m)]))# Rename
       columns
332    df.m %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, m=
       value) %>% mutate(time=gsub(x=time, pattern="t", replacement="")
       , time=as.numeric(time)) %>% arrange(time,Y,x) -> df.m#
       Additional columns containing info on input
333
334    # 2.2 --- Oligomer density data frame
335    data.frame(x=rep(seq(0,1,Delta.x), Nx), Y=sort(rep(seq(0,1,Delta.
       x), Nx)),p) -> df.p# Turn the matrix into a data frame.
336    names(df.p)=c("x","Y", paste0("t",time.vals[1:ncol(p)]))# Rename
       columns
337    df.p %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, p=
       value) %>% mutate(time=gsub(x=time, pattern="t", replacement="")
       , time=as.numeric(time)) %>% arrange(time,Y,x) -> df.p#
       Additional columns containing info on input
338
339    # 2.3 --- Polymer density data frame
```

```
340   data.frame(x=rep(seq(0,1,Delta.x), Nx), Y=sort(rep(seq(0,1,Delta.
       x), Nx)), q) -> df.q# Turn the matrix into a data frame.
341   names(df.q)=c("x","Y", paste0("t",time.vals[1:ncol(q)]))# Rename
       columns
342   df.q %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, q=
       value) %>% mutate(time=gsub(x=time, pattern="t", replacement="")
       , time=as.numeric(time)) %>% arrange(time,Y,x) -> df.q#
       Additional columns containing info on input
343
344   # 2.4 --- Zeolite density data frame
345   data.frame(x=rep(seq(0,1,Delta.x), Nx), Y=sort(rep(seq(0,1,Delta.
       x), Nx)), z) -> df.z# Turn the matrix into a data frame.
346   names(df.z)=c("x","Y", paste0("t",time.vals[1:ncol(z)]))# Rename
       columns
347   df.z %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, z=
       value) %>% mutate(time=gsub(x=time, pattern="t", replacement="")
       , time=as.numeric(time),z=z0*z) %>% arrange(time,Y,x) -> df.z#
       Additional columns containing info on input
348
349   data.frame(x=rep(seq(0,1,Delta.x), Nx), Y=sort(rep(seq(0,1,Delta.
       x), Nx)), b) -> df.b# Turn the matrix into a data frame.
350   names(df.b)=c("x","Y", paste0("t",time.vals[1:ncol(b)]))# Rename
       columns
351   df.b %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, b=
       value) %>% mutate(time=gsub(x=time, pattern="t", replacement="")
       , time=as.numeric(time),b=b0*b) %>% arrange(time,Y,x) -> df.b#
       Additional columns containing info on input
352
353   # 3.1 --- Refractive index matrix
354   Vb = b*b0/rhob # cm**3
355   Vm = m*m0/rhom # cm**3
356   Vp = p*m0/rhop # cm**3
357   Vq = q*m0/rhop # cm**3
358   Vz = z*z0/rhoz # cm**3
359   Vtotal=Vb+Vm+Vp+Vq+Vz# cm**3
360
361   phi.m=Vm/Vtotal
362   phi.p=Vp/Vtotal
363   phi.q=Vq/Vtotal
364   phi.z=Vz/Vtotal
365   phi.b=Vb/Vtotal
366
367   Lorentz.Lorenz.RHS = phi.m*(nm*nm - 1)/(nm*nm + 2) + phi.b*(nb*nb
       - 1)/(nb*nb + 2) + phi.p*(np*np - 1)/(np*np + 2) + phi.q*(nq*nq
       - 1)/(nq*nq + 2) + phi.z*(nz*nz - 1)/(nz*nz + 2)
368
369   n=sqrt((2*Lorentz.Lorenz.RHS + 1)/(1 - Lorentz.Lorenz.RHS))
370
371   data.frame(x=rep(seq(0,1,Delta.x), Nx), Y=sort(rep(seq(0,1,Delta.
       x), Nx)), n) -> df.n# Turn the matrix into a data frame.
```

```r
372    names(df.n)=c("x","Y", paste0("t",time.vals[1:ncol(n)]))# Rename
         columns
373    df.n %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, n=
         value) %>% mutate(time=gsub(x=time, pattern="t", replacement="")
         , time=as.numeric(time)) %>% arrange(time,Y,x) -> df.n#
         Additional columns containing info on input
374
375    cbind(df.b,m=df.m$m,p=df.p$p,q=df.q$q,z=df.z$z,n=df.n$n) -> df1
376
377    # 4.1 --- Refractive index modulation data frame
378    Nt = length(time)
379    N0=matrix(0, nrow=Nt, ncol=Nx)
380    N1a=matrix(0, nrow=Nt, ncol=Nx)
381    N1b=matrix(0, nrow=Nt, ncol=Nx)
382    d2=matrix(0, nrow=Nt, ncol=Nx)
383    int_n2=matrix(0, nrow=Nt, ncol=Nx)
384    Mean.RI=rep(0,Nt)
385
386    for(k in 1:Nt){
387
388      matrix(n[, k], nrow=Nx) -> n1
389
390      n2=n1**2
391
392      Mean.RI[k]=Delta.x*Delta.x/4*(n1[1,1] + n1[Nx,1] + n1[1,Nx] +
         n1[Nx,Nx] + sum(2*n1[1,(2:(Nx-1))]) + sum(2*n1[(2:(Nx-1)),1]) +
         sum(2*n1[Nx,(2:(Nx-1))]) + sum(2*n1[(2:(Nx-1)),Nx]) + sum(4*n1
         [(2:(Nx-1)),(2:(Nx-1))]) )
393
394      for(i in 1:Nx){
395
396        n1[, i] -> n1i
397        N0[k, i]=Delta.x/3*(n1i[1] + sum(2*n1i[times.2]) + sum(4*n1i[
         times.4]) + n1i[Nx])
398
399        n1[, i]*cos(2*pi*x) -> n1cos
400        N1a[k, i]=2*Delta.x/3*(n1cos[1] + sum(2*n1cos[times.2]) + sum
         (4*n1cos[times.4]) + n1cos[Nx])
401
402        n1[, i]*sin(2*pi*x) -> n1sin
403        N1b[k, i]=2*Delta.x/3*(n1sin[1] + sum(2*n1sin[times.2]) + sum
         (4*n1sin[times.4]) + n1sin[Nx])
404
405      }
406
407
408      n.Fourier=matrix(0, nrow=Nx, ncol=Nx)
409
410      for(i in 1:Nx){
411
```

```r
412    n.Fourier[,i] = N0[k, i] + N1a[k, i]*cos(2*pi*x) + N1b[k, i]*
    sin(2*pi*x)

413
414    }
415
416    diff = abs(n1 - n.Fourier)**2
417
418    for(i in 1:Nx){
419      d2[k,i] = Delta.x/3*(diff[1, i] + sum(2*diff[times.2, i]) +
    sum(4*diff[times.4, i]) + diff[Nx, i])
420      int_n2[k,i] = Delta.x/3*(n2[1, i] + sum(2*n2[times.2, i]) +
    sum(4*n2[times.4, i]) + n2[Nx, i])
421    }
422
423  }# End for loop
424
425  delta=d2/int_n2
426  Delta.n=2*sqrt(N1a*N1a + N1b*N1b)
427
428  data.frame(
429    time=rep(seq(0,total.time,output.time.step), Nx),
430    Y=sort(rep(seq(0,1,Delta.x), Nt)),
431    Delta.n=melt(Delta.n)$value,
432    delta=melt(delta)$value
433  ) -> df2
434
435  df1 %>% mutate(Delta.n=0, delta=0) %>% arrange(time,Y,x) -> df1
436
437  for(i in time){
438    for(j in Y){
439      df1[df1$time==i & df1$Y==j, "Delta.n"] = df2[df2$time==i &
    df2$Y==j, "Delta.n"]
440      df1[df1$time==i & df1$Y==j, "delta"] = df2[df2$time==i & df2$
    Y==j, "delta"]
441    }
442  }
443
444  data.frame(
445    time,
446    T.t=T0*u.t,
447    Mean.RI,
448    Actual.Shrinkage,
449    Lambda.t,
450    phi.r.t,
451    int_M,
452    int_P,
453    int_Q,
454    int_Z,
455    int_B
456  ) %>% mutate(
```

```
457      theta_B=asin(lambda.probe/2/Mean.RI/Lambda.t) - phi.r.t,
458      Delta_theta_B=theta_B0 - theta_B,
459      Delta_phi_r=phi.r0 - phi.r.t,
460      Apparent.Shrinkage = 1 - tan(phi.r0)/tan(phi.r0 + Delta_theta_B
       )
461    ) -> df3
462
463    for(i in time){
464      for(j in c("T.t","Mean.RI","Actual.Shrinkage","Lambda.t","phi.r
       .t","int_M","int_P","int_Q","int_Z","int_B","theta_B","Delta_
       theta_B","Delta_phi_r","Apparent.Shrinkage")){
465        df1[df1$time==i, j] = df3[df3$time==i, j]
466      }
467    }
468
469    df1 %>%
470      mutate(
471        time=as.numeric(time),
472        Y=as.numeric(Y),
473        start_exp,
474        end_exp,
475        total.time,
476        lpmm,
477        I0,
478        slant.angle=ifelse(abs(slant.angle) < 1e-3, 0, slant.angle),
479        xi,
480        Dm,
481        Dp,
482        epsilon.pz,
483        epsilon.qz,
484        nm,
485        rhom,
486        rhop,
487        np=np,
488        nq=nq,
489        Gamma,
490        wt.pc,
491        Dz,
492        rhoz,
493        nz,
494        nb,
495        rhob,
496        b0,
497        z0,
498        T0,
499        zeta,
500        lambda.probe,
501        nu=pi*Delta.n*T.t/lambda.probe/cos(theta_B),
502        Moharam.Young=lambda.probe*lambda.probe/Mean.RI/Delta.n/
       Lambda.t/Lambda.t/cos(phi.r.t),
```

```r
      Klein.Cook=2*pi*lambda.probe*T.t/Mean.RI/Lambda.t/Lambda.t/
    cos(phi.r.t),
      Delta.t,
      Delta.x,
      output.time.step,
      Model="slanted_grating_simulation_v21"
    ) %>%
    arrange(time,Y,x) -> df1

  if(df1 %>% subset(Y==0 & time==max(time)) %>% distinct(Klein.Cook
    ) %>% pull(Klein.Cook) < 10){

    df1 %>% mutate(J0=nu/2, J1=J0) -> df1

    for(l in 1:100){

      df1$J1 = df1$J1 + ((-1)**l)/factorial(l)/factorial(l+1)*df1$
    J0**(2*l + 1)

    }

    df1 %>% mutate(

      eta=J1*J1,
      Geometry="Planar"

    ) %>% select(-J1,-J0) -> df1

  } else {

      df1 %>% mutate(

        eta=sin(sqrt(nu*nu))**2,
        Geometry="Volume"

      ) -> df1

  }

  if(
    nrow(
      subset(
        df1,
        z < 0
      )
    ) > 0
  ){
    return(warning("Fail: z(x,t) < 0"))
  } else {
    return(df1)
```

```
550     }
551
552 }
```

Holographic Sensor

```r
 1 sensor_1D_model_v1 = function(
 2
 3   sim_holo_grat=NULL,
 4   ns=1.33,
 5   na=2,
 6   nzs=1.46,
 7   nza=1.6,
 8   tau_c_s=60,# seconds
 9   tau_c_a=1,# seconds
10   tau_e_s=60,# seconds
11   tau_e_a=60,# seconds
12   a0=0.1*0.1,# m0
13   s0=0.1,# m0
14   rhoa=9,# g/cm3
15   rhos=1,# g/cm3
16   Da=2.3e-5,# g/cm3
17   Ds=2.3e-5,# cm2/s
18   lambda_probe=633e-7,# cm
19   exposure_time=180,# seconds
20   output_time_step=5# seconds
21
22 ){
23
24   if(is.null(sim_holo_grat)){
25     return(warning("Function needs a simulation of a holographic
    grating"))
26   }
27
28   # if(!(unique(sim_holo_grat$Model)=="slanted_grating_simulation_
    v16")){
29   #   return(warning("Theoretically modelled nanocomposite needs to
    be simulated with slanted_grating_simulation_v16()."))
30   # }
31
32   # 1.2 --- Define the holographic grating parameters
33   sim_holo_grat %>% pull(Delta.x) %>% unique() -> Delta_x
34   sim_holo_grat %>% pull(Delta.t) %>% unique() -> Delta_t
35   r = Delta_t/Delta_x/Delta_x
36   Nx=1/Delta_x + 1# Number of spatial points
37
38   interior_points = 2:(Nx-1)
39   times_4 = interior_points[interior_points %% 2 == 0]
40   times_2 = interior_points[interior_points %% 2 != 0]
```

```r
df0=subset(sim_holo_grat, time==max(time))

df0 %>% pull(T.t) %>% unique() -> T1

df0 %>% pull(lpmm) %>% unique() -> lpmm

df0 %>% pull(b0) %>% unique() -> b0

Lambda=1/10/lpmm

df0 %>% pull(nz) %>% unique() -> nze

#LLRHS = phi_interior*((ns*ns - 1)/(ns*ns + 2)) + ((nze*nze - 1)
    /(nze*nze + 2))

#nzs = sqrt((2*LLRHS + 1)/(1 - LLRHS))

df0 %>% pull(wt.pc) %>% unique() -> wt_pc

t0=1

m0=1

sim_holo_grat %>% pull(z0) %>% unique() -> z0

if(z0==0){z0=1e-6}
if(tau_c_s==0){tau_c_s=1e-4}
if(tau_e_s==0){tau_e_s=1e-4}

gamma_s=1/z0/tau_c_s
gamma_a=1/z0/tau_c_a
omega_s=s0/z0/tau_e_s
omega_a=a0/z0/tau_e_a

df0 %>% arrange(x) %>% pull(n) -> n_before_exposure
df0 %>% pull(nb) %>% unique -> nb
df0 %>% pull(nm) %>% unique -> nm
df0 %>% pull(np) %>% unique -> np
df0 %>% pull(nq) %>% unique -> nq
df0 %>% pull(nz) %>% unique -> nze

rep(1, Nx) -> b1
df0 %>% arrange(x) %>% pull(m) -> m1
df0 %>% arrange(x) %>% pull(p) -> p1
df0 %>% arrange(x) %>% pull(q) -> q1
df0 %>% arrange(x) %>% mutate(z1=ifelse(z0==0, z, z/z0)) %>% pull
    (z1) -> ze1
rep(1, Nx) -> s1
rep(1, Nx) -> a1
```

```r
89    rep(0, Nx) -> zs1
90    rep(0, Nx) -> za1
91
92    matrix(a1, ncol=1) -> a
93    matrix(s1, ncol=1) -> s
94    matrix(b1, ncol=1) -> b
95    matrix(m1, ncol=1) -> m
96    matrix(p1, ncol=1) -> p
97    matrix(q1, ncol=1) -> q
98    matrix(ze1, ncol=1) -> ze
99    matrix(zs1, ncol=1) -> zs
100   matrix(za1, ncol=1) -> za
101
102   int_m_dx=Delta_x/3*(m1[1] + sum(2*m1[times_2]) + sum(4*m1[times
      _4]) + m1[Nx])
103   int_p_dx=Delta_x/3*(p1[1] + sum(2*p1[times_2]) + sum(4*p1[times
      _4]) + p1[Nx])
104   int_q_dx=Delta_x/3*(q1[1] + sum(2*q1[times_2]) + sum(4*q1[times
      _4]) + q1[Nx])
105
106   sim_holo_grat %>% pull(rhom) %>% unique() -> rhom
107   sim_holo_grat %>% pull(rhop) %>% unique() -> rhop
108   sim_holo_grat %>% pull(rhob) %>% unique() -> rhob
109   sim_holo_grat %>% pull(rhoz) %>% unique() -> rhoz
110
111   v1=b0/rhob + int_m_dx/rhom + int_p_dx/rhop + int_q_dx/rhop + z0/
      rhoz
112   v2=s0/rhos + a0/rhoa + v1
113   T2=c(T1*v2/v1)
114
115   alpha_s_x=Ds*t0/Lambda/Lambda
116   alpha_a_x=Da*t0/Lambda/Lambda
117   gamma_ss=gamma_s*z0*t0
118   omega_ss=ifelse(s0==0, 0, omega_s*z0*t0/s0)
119   gamma_sz=gamma_s*s0*t0
120   omega_sz=omega_s*t0
121   gamma_aa=gamma_a*z0*t0
122   omega_aa=ifelse(a0==0, 0, omega_a*z0*t0/a0)
123   gamma_az=gamma_a*a0*t0
124   omega_az=omega_a*t0
125
126   Nt = exposure_time/Delta_t+1# Total number of iterations
127
128   seq(0, exposure_time, by=output_time_step) -> time_vals
129
130   if(exposure_time < output_time_step){
131     bind_iter="0"
132   } else {
133     time_vals/Delta_t + 1 -> bind_iter
134     bind_iter[-1] -> bind_iter
```

```r
    }

    # Simulation of holographic grating exposed to a loaded solvent
    for(j in 1:Nt){

        AA2 = diag(2 + 2*r*alpha_a_x + gamma_aa*Delta_t*as.numeric(
    ze1), Nx)
        AA1 = diag(2 - 2*r*alpha_a_x - gamma_aa*Delta_t*as.numeric(
    ze1), Nx)

        SS2 = diag(2 + 2*r*alpha_s_x + gamma_ss*Delta_t*as.numeric(
    ze1), Nx)
        SS1 = diag(2 - 2*r*alpha_s_x - gamma_ss*Delta_t*as.numeric(
    ze1), Nx)

        ZEZE2 = diag(2 + gamma_sz*Delta_t*as.numeric(s1) + gamma_az*
    Delta_t*as.numeric(a1), Nx)
        ZEZE1 = diag(2 - gamma_sz*Delta_t*as.numeric(s1) - gamma_az*
    Delta_t*as.numeric(a1), Nx)

        ZSZS2 = diag(2 + omega_sz*Delta_t, Nx)
        ZSZS1 = diag(2 - omega_sz*Delta_t, Nx)

        ZAZA2 = diag(2 + omega_az*Delta_t, Nx)
        ZAZA1 = diag(2 - omega_az*Delta_t, Nx)

        for(i in 1:Nx){

            i_minus_1=ifelse(i == 1, i + 1, i - 1)
            i_plus_1=ifelse(i == Nx, i - 1, i + 1)

            SS2[i, i_minus_1] = SS2[i, i_minus_1] - r*alpha_s_x
            SS2[i, i_plus_1] = SS2[i, i_plus_1] - r*alpha_s_x

            SS1[i, i_minus_1] = SS1[i, i_minus_1] + r*alpha_s_x
            SS1[i, i_plus_1] = SS1[i, i_plus_1] + r*alpha_s_x

            AA2[i, i_minus_1] = AA2[i, i_minus_1] - r*alpha_a_x
            AA2[i, i_plus_1] = AA2[i, i_plus_1] - r*alpha_a_x

            AA1[i, i_minus_1] = AA1[i, i_minus_1] + r*alpha_a_x
            AA1[i, i_plus_1] = AA1[i, i_plus_1] + r*alpha_a_x

        }

        s2 = solve(SS2) %*% ((SS1 %*% matrix(s1, ncol=1)) + matrix(2*
    Delta_t*omega_ss*zs1, ncol=1))
        a2 = solve(AA2) %*% ((AA1 %*% matrix(a1, ncol=1)) + matrix(2*
    Delta_t*omega_aa*za1, ncol=1))
```

```r
176      ze2 = solve(ZEZE2) %*% ((ZEZE1 %*% matrix(ze1, ncol=1)) +
      matrix(2*omega_sz*Delta_t*zs1, ncol=1) + matrix(2*omega_az*Delta
      _t*za1, ncol=1))
177      zs2 = solve(ZSZS2) %*% ((ZSZS1 %*% matrix(zs1, ncol=1)) +
      matrix(2*Delta_t*gamma_sz*s1*ze1, ncol=1))
178      za2 = solve(ZAZA2) %*% ((ZAZA1 %*% matrix(za1, ncol=1)) +
      matrix(2*Delta_t*gamma_az*a1*ze1, ncol=1))
179
180      s2 -> s1
181      a2 -> a1
182      ze2 -> ze1
183      zs2 -> zs1
184      za2 -> za1
185
186      int_s_dx=Delta_x/3*(s1[1] + sum(2*s1[times_2]) + sum(4*s1[
      times_4]) + s1[Nx])
187      int_a_dx=Delta_x/3*(a1[1] + sum(2*a1[times_2]) + sum(4*a1[
      times_4]) + a1[Nx])
188
189      v2 = v1 + s0*int_s_dx/rhos + a0*int_a_dx/rhoa
190
191      if(as.character(j) %in% as.character(bind_iter)){
192
193          matrix(cbind(s, s1), nrow=Nx) -> s
194          matrix(cbind(a, a1), nrow=Nx) -> a
195          matrix(cbind(ze, ze1), nrow=Nx) -> ze
196          matrix(cbind(zs, zs1), nrow=Nx) -> zs
197          matrix(cbind(za, za1), nrow=Nx) -> za
198          T2 = c(T2, T1*v2/v1)
199
200      }
201
202  }
203
204  # Mass concentration data frame
205  x=seq(0,1,Delta_x)
206  time=seq(0,exposure_time,output_time_step)
207  Nt=length(time)
208
209  data.frame(x=rep(x, Nt), time=sort(rep(time, Nx)), s=s0*melt(s)$
      value, a=a0*melt(a)$value, ze=z0*melt(ze)$value, zs=z0*melt(zs)$
      value, za=z0*melt(za)$value, m=rep(m1, Nt), p=rep(p1, Nt), q=rep
      (q1, Nt), b=rep(b0, Nt), n_before_exposure=rep(n_before_exposure
      , Nt)) -> df2
210
211  df2 %>%
212
213   mutate(
214
215     rhoz=rhoz,
```

```
216        rhos=rhos,
217        rhoa=rhoa,
218        nze=nze,
219        nzs=nzs,
220        nza=nza,
221        ns=ns,
222        na=na,
223
224        Vm=m/rhom,
225        Vp=p/rhop,
226        Vq=q/rhop,
227        Vb=b/rhob,
228        Vze=ze/rhoz,
229        Vzs=zs/rhoz,
230        Vza=za/rhoz,
231        Vs=s/rhos,
232        Va=a/rhoa,
233
234        V=Vb+Vm+Vp+Vq+Vze+Vzs+Vza+Vs+Va,
235
236        phi_m=Vm/V,
237        phi_p=Vp/V,
238        phi_q=Vq/V,
239        phi_b=Vb/V,
240        phi_ze=Vze/V,
241        phi_zs=Vzs/V,
242        phi_za=Vza/V,
243        phi_s=Vs/V,
244        phi_a=Va/V,
245
246        Lorentz_Lorenz_RHS = phi_m*(nm*nm - 1)/(nm*nm + 2) + phi_p*(
    np*np - 1)/(np*np + 2) + phi_q*(nq*nq - 1)/(nq*nq + 2) + phi_b*(
    nb*nb - 1)/(nb*nb + 2) + phi_ze*(nze*nze - 1)/(nze*nze + 2) +
    phi_zs*(nzs*nzs - 1)/(nzs*nzs + 2) + phi_za*(nza*nza - 1)/(nza*
    nza + 2) + phi_s*(ns*ns - 1)/(ns*ns + 2) + phi_a*(na*na - 1)/(na
    *na + 2),
247
248        n_after_exposure=sqrt((2*Lorentz_Lorenz_RHS + 1)/(1 - Lorentz
    _Lorenz_RHS))
249
250    ) %>% select(-Lorentz_Lorenz_RHS,-Vb,-Vm,-Vp,-Vq,-Vze,-Vzs,-Vs
    ,-V,-phi_b,-phi_m,-phi_p,-phi_q,-phi_ze,-phi_zs,-phi_s,-phi_a,-
    phi_za,-Va,-Vza) %>% arrange(time,x) -> df3
251
252    #Refractive index modulation data frame
253    n = df3 %>% arrange(time,x) %>% pull(n_after_exposure) %>% matrix
    (nrow=Nx)
254
255    N0=rep(0, Nt)
256    N1=rep(0, Nt)
```

```r
257   n.tilde=matrix(0, nrow=nrow(n), ncol=ncol(n))
258   diff=matrix(0, nrow=nrow(n), ncol=ncol(n))
259   n2=matrix(0, nrow=nrow(n), ncol=ncol(n))
260   d2=rep(0, Nt)
261   int.n2=rep(0,Nt)
262
263   for(k in 1:Nt){# For each time point ...
264
265     interior.points = 2:(Nx-1)
266     times.4 = interior.points[interior.points %% 2 == 0]
267     times.2 = interior.points[interior.points %% 2 != 0]
268
269     n0=n[, k]
270     n1=n[, k]*cos(2*pi*x)
271
272     N0[k]=Delta_x/3*(n0[1] + n0[Nx] + sum(2*n0[times.2]) + sum(4*n0
      [times.4]))
273     N1[k]=2*Delta_x/3*(n1[1] + n1[Nx] + sum(2*n1[times.2]) + sum(4*
      n1[times.4]))
274
275     n.tilde[, k] = N0[k] + N1[k]*cos(2*pi*x)
276     diff[, k] = (n[, k] - n.tilde[, k])**2
277     d2[k] = Delta_x/3*(diff[1, k] + diff[Nx, k] + sum(2*diff[times
      .2, k]) + sum(4*diff[times.4, k]))
278     n2[, k]=n[, k]**2
279     int.n2[k]=Delta_x/3*(n2[1, k] + sum(2*n2[times.2, k]) + sum(4*
      n2[times.4, k]) + n2[Nx, k])
280
281   }# End for loop
282
283   df3 %>% arrange(time,x) %>% mutate(Delta_n=0, distortion=0, Mean_
      RI=0, theta_B=0) -> df4
284
285   for(i in x){
286     df4[df4$x==i, "Delta_n"]=2*N1
287     df4[df4$x==i, "distortion"]=d2/int.n2
288     df4[df4$x==i, "Mean_RI"]=N0
289     df4[df4$x==i, "T2"]=T2
290   }
291
292   df4 %>% mutate(
293     theta_B=asin(lambda_probe/2/Mean_RI/Lambda),
294     Moharam.Young=lambda_probe*lambda_probe/Mean_RI/Delta_n/Lambda/
      Lambda/cos(0),
295     Klein.Cook=2*pi*lambda_probe*T2/Mean_RI/Lambda/Lambda/cos(0)
296   ) -> df4
297
298   if(df4 %>% subset(time==0) %>% pull(Klein.Cook) %>% unique < 10){
299
300     df4 %>% mutate(J0=pi*Delta_n*T2/lambda_probe/2, J1=J0) -> df4
```

```r
301
302    for(l in 1:100){
303
304       df4$J1 = df4$J1 + ((-1)**l)/factorial(l)/factorial(l+1)*df4$
    J0**(2*l + 1)
305
306    }
307
308    df4 %>% mutate(
309
310       eta=J1*J1,
311       Geometry="Planar"
312
313    ) %>% select(-J1,-J0) -> df5
314
315  } else {
316
317       df4 %>% mutate(
318
319          Delta_phi_r=0,
320          nu=pi*Delta_n*T2/lambda_probe/cos(theta_B),
321          xi=pi*T2*Delta_phi_r/Lambda,
322          eta=sin(sqrt(xi*xi + nu*nu))**2/sqrt(1 + (xi*xi)/(nu*nu)),
323          Geometry="Volume"
324
325       ) -> df5
326
327  }
328
329  df5 %>% subset(time==0) %>% pull(eta) %>% unique -> pre_exposure_
    eta
330  df5 %>% subset(time==0) %>% pull(theta_B) %>% unique -> pre_
    exposure_Bragg_angle
331
332  df5 %>% mutate(
333
334    time=as.numeric(time),
335    x=as.numeric(x),
336    pre_exposure_eta,
337    pre_exposure_Bragg_angle,
338    lambda_r_t=2*Mean_RI*Lambda*sin(theta_B),
339    normalized_eta=eta/pre_exposure_eta,
340    Delta_theta_B=theta_B-pre_exposure_Bragg_angle,
341    nze,nzs,ns,z0,s0,a0,b0,tau_c_s,tau_e_s,tau_c_a,tau_e_a,
342    tau_x_s=Lambda*Lambda/Ds,tau_x_a=Lambda*Lambda/Da,
343    gamma_s,omega_s,rhoz,rhos,Ds,wt_pc,
344    T1, Delta_T=T2-T1,
345    lpmm,exposure_time,output_time_step,
346    model="sensor_1D_model_v1"
347  ) %>% arrange(time,x) -> df6
```

```
348
349   return(df6)
350
351 }
```

```
1 sensor_2D_model_v2 = function(
2
3   sim_holo_grat=NULL,
4   ns=1.33,
5   na=2,
6   nzs=1.46,
7   nza=1.6,
8   tau_c_s=60,# seconds
9   tau_c_a=1,# seconds
10  tau_e_s=60,# seconds
11  tau_e_a=60,# seconds
12  a0=0.1*0.1,# m0
13  s0=0.1,# m0
14  rhoa=9,# g/cm3
15  rhos=1,# g/cm3
16  Da=2.3e-5,# cm/2
17  Ds=2.3e-5,# cm2/s
18  lambda_probe=633e-7,# cm
19  exposure_time=180,# seconds
20  output_time_step=5# seconds
21
22 ){
23
24   if(is.null(sim_holo_grat)){
25     return(warning("Function needs a simulation of a holographic
       grating"))
26   }
27
28   # if(!(unique(sim_holo_grat$Model)=="slanted_grating_simulation_
       v16")){
29   #   return(warning("Theoretically modelled nanocomposite needs to
         be simulated with slanted_grating_simulation_v16()."))
30   # }
31
32   if(unique(sim_holo_grat$slant.angle)==0){sim_holo_grat$slant.
       angle=1e-4}
33
34   # 1.2 --- Define the holographic grating parameters
35   sim_holo_grat %>% pull(Delta.x) %>% unique() -> Delta_x
36   sim_holo_grat %>% pull(Delta.t) %>% unique() -> Delta_t
37   r = Delta_t/Delta_x/Delta_x
38   Nx=1/Delta_x + 1# Number of spatial points
39
40   interior_points = 2:(Nx-1)
41   times_4 = interior_points[interior_points %% 2 == 0]
42   times_2 = interior_points[interior_points %% 2 != 0]
```

```r
43
44   df0=subset(sim_holo_grat, time==max(time))
45
46   df0 %>% pull(lpmm) %>% unique() -> lpmm
47
48   df0 %>% pull(b0) %>% unique() -> b0
49
50   df0 %>% mutate(x_hat=Lambda.t/cos(phi.r.t)) %>% pull(x_hat) %>%
      unique() -> x_hat
51
52   df0 %>% mutate(y_hat=Lambda.t/sin(phi.r.t)) %>% pull(y_hat) %>%
      unique() -> y_hat_1
53
54   df0 %>% pull(T.t) %>% unique() -> T_1
55
56   df0 %>% pull(Lambda.t) %>% unique() -> Lambda_1
57
58   df0 %>% pull(phi.r.t) %>% unique() -> phi_r_1
59
60   df0 %>% pull(theta_B) %>% unique() -> theta_B1
61
62   df0 %>% mutate(lambda_r_1=2*Mean.RI*Lambda.t*sin(phi.r.t)) %>%
      pull(lambda_r_1) %>% unique() -> lambda_r_1
63
64   df0 %>% pull(nz) %>% unique() -> nze
65
66   #LLRHS = phi_interior*((ns*ns - 1)/(ns*ns + 2)) + ((nze*nze - 1)
      /(nze*nze + 2))
67
68   #nzs = sqrt((2*LLRHS + 1)/(1 - LLRHS))
69
70   df0 %>% pull(wt.pc) %>% unique() -> wt_pc
71
72   t0=1
73
74   m0=1
75
76   sim_holo_grat %>% pull(z0) %>% unique() -> z0
77
78   if(z0==0){z0=1e-6}
79   if(tau_c_s==0){tau_c_s=1e-4}
80   if(tau_e_s==0){tau_e_s=1e-4}
81   if(tau_c_a==0){tau_c_a=1e-4}
82   if(tau_e_a==0){tau_e_a=1e-4}
83
84   gamma_a=1/z0/tau_c_a
85   gamma_s=1/z0/tau_c_s
86   omega_a=a0/z0/tau_e_a
87   omega_s=s0/z0/tau_e_s
88
```

```
89   df0 %>% arrange(x) %>% pull(n) -> n_before_exposure
90   df0 %>% pull(nb) %>% unique -> nb
91   df0 %>% pull(nm) %>% unique -> nm
92   df0 %>% pull(np) %>% unique -> np
93   df0 %>% pull(nq) %>% unique -> nq
94   df0 %>% pull(nz) %>% unique -> nze
95   free_surface=(Nx*Nx - Nx + 1):(Nx*Nx)
96   fixed_surface=1:Nx
97
98   rep(1, Nx*Nx) -> b1
99   df0 %>% arrange(Y,x) %>% pull(m) -> m1
100  df0 %>% arrange(Y,x) %>% pull(p) -> p1
101  df0 %>% arrange(Y,x) %>% pull(q) -> q1
102  df0 %>% arrange(Y,x) %>% mutate(z1=ifelse(z0==0, z, z/z0)) %>%
      pull(z1) -> ze1
103  rep(0, Nx*Nx) -> s1; s1[free_surface]=1
104  rep(0, Nx*Nx) -> a1; a1[free_surface]=1
105  rep(0, Nx*Nx) -> zs1
106  rep(0, Nx*Nx) -> za1
107
108  matrix(a1, ncol=1) -> a
109  matrix(s1, ncol=1) -> s
110  matrix(b1, ncol=1) -> b
111  matrix(m1, ncol=1) -> m
112  matrix(p1, ncol=1) -> p
113  matrix(q1, ncol=1) -> q
114  matrix(ze1, ncol=1) -> ze
115  matrix(zs1, ncol=1) -> zs
116  matrix(za1, ncol=1) -> za
117
118  m1 %>% matrix(nrow=Nx) -> matrix.m
119  p1 %>% matrix(nrow=Nx) -> matrix.p
120  q1 %>% matrix(nrow=Nx) -> matrix.q
121  a1 %>% matrix(nrow=Nx) -> matrix.a
122  s1 %>% matrix(nrow=Nx) -> matrix.s
123
124  int_m_dx_dy=Delta_x*Delta_x/4*(matrix.m[1,1] + matrix.m[Nx,1] +
      matrix.m[1,Nx] + matrix.m[Nx,Nx] + sum(2*matrix.m[1,(2:(Nx-1))])
       + sum(2*matrix.m[(2:(Nx-1)),1]) + sum(2*matrix.m[Nx,(2:(Nx-1))
      ]) + sum(2*matrix.m[(2:(Nx-1)),Nx]) + sum(4*matrix.m[(2:(Nx-1))
      ,(2:(Nx-1))]) )
125
126  int_p_dx_dy=Delta_x*Delta_x/4*(matrix.p[1,1] + matrix.p[Nx,1] +
      matrix.p[1,Nx] + matrix.p[Nx,Nx] + sum(2*matrix.p[1,(2:(Nx-1))])
       + sum(2*matrix.p[(2:(Nx-1)),1]) + sum(2*matrix.p[Nx,(2:(Nx-1))
      ]) + sum(2*matrix.p[(2:(Nx-1)),Nx]) + sum(4*matrix.p[(2:(Nx-1))
      ,(2:(Nx-1))]) )
127
128  int_q_dx_dy=Delta_x*Delta_x/4*(matrix.q[1,1] + matrix.q[Nx,1] +
      matrix.q[1,Nx] + matrix.q[Nx,Nx] + sum(2*matrix.q[1,(2:(Nx-1))])
```

```
     + sum(2*matrix.q[(2:(Nx-1)),1]) + sum(2*matrix.q[Nx,(2:(Nx-1))
     ]) + sum(2*matrix.q[(2:(Nx-1)),Nx]) + sum(4*matrix.q[(2:(Nx-1))
     ,(2:(Nx-1))]) )

129

130  int_a_dx_dy=Delta_x*Delta_x/4*(matrix.a[1,1] + matrix.a[Nx,1] +
     matrix.a[1,Nx] + matrix.a[Nx,Nx] + sum(2*matrix.a[1,(2:(Nx-1))])
     + sum(2*matrix.a[(2:(Nx-1)),1]) + sum(2*matrix.a[Nx,(2:(Nx-1))
     ]) + sum(2*matrix.a[(2:(Nx-1)),Nx]) + sum(4*matrix.a[(2:(Nx-1))
     ,(2:(Nx-1))]) )

131

132  int_s_dx_dy=Delta_x*Delta_x/4*(matrix.s[1,1] + matrix.s[Nx,1] +
     matrix.s[1,Nx] + matrix.s[Nx,Nx] + sum(2*matrix.s[1,(2:(Nx-1))])
     + sum(2*matrix.s[(2:(Nx-1)),1]) + sum(2*matrix.s[Nx,(2:(Nx-1))
     ]) + sum(2*matrix.s[(2:(Nx-1)),Nx]) + sum(4*matrix.s[(2:(Nx-1))
     ,(2:(Nx-1))]) )

133

134  sim_holo_grat %>% pull(rhom) %>% unique() -> rhom
135  sim_holo_grat %>% pull(rhop) %>% unique() -> rhop
136  sim_holo_grat %>% pull(rhob) %>% unique() -> rhob
137  sim_holo_grat %>% pull(rhoz) %>% unique() -> rhoz

138

139  volume1=b0/rhob + int_m_dx_dy/rhom + int_p_dx_dy/rhop + int_q_dx_
     dy/rhop + z0/rhoz + a0*int_a_dx_dy/rhoa + s0*int_s_dx_dy/rhos

140

141  # 3.1 --- Diffusion of target a
142  alpha_s_x=Ds*t0/x_hat/x_hat
143  alpha_a_x=Da*t0/x_hat/x_hat
144  alpha_s_y=Ds*t0/T_1/T_1
145  alpha_a_y=Da*t0/T_1/T_1
146  gamma_ss=gamma_s*z0*t0
147  omega_ss=ifelse(s0==0, 0, omega_s*z0*t0/s0)
148  gamma_sz=gamma_s*s0*t0
149  omega_sz=omega_s*t0
150  gamma_aa=gamma_a*z0*t0
151  omega_aa=ifelse(a0==0, 0, omega_a*z0*t0/a0)
152  gamma_az=gamma_a*a0*t0
153  omega_az=omega_a*t0

154

155  n_iterations = exposure_time/Delta_t + 1# Total number of
     iterations

156

157  seq(0, exposure_time, by=output_time_step) -> time_vals

158

159  if(exposure_time < output_time_step){
160    bind_iter="0"
161  } else {
162    time_vals/Delta_t + 1 -> bind_iter
163    bind_iter[-1] -> bind_iter
164  }

165
```

```r
166   # Simulation of holographic grating exposed to a loaded solvent
167   for(j in 1:n_iterations){
168
169     AA2 = diag(2 + 2*r*alpha_a_x + 2*r*alpha_a_y + gamma_aa*Delta_t
      *as.numeric(ze1), Nx*Nx)
170     AA1 = diag(2 - 2*r*alpha_a_x - 2*r*alpha_a_y - gamma_aa*Delta_t
      *as.numeric(ze1), Nx*Nx)
171
172     SS2 = diag(2 + 2*r*alpha_s_x + 2*r*alpha_s_y + gamma_ss*Delta_t
      *as.numeric(ze1), Nx*Nx)
173     SS1 = diag(2 - 2*r*alpha_s_x - 2*r*alpha_s_y - gamma_ss*Delta_t
      *as.numeric(ze1), Nx*Nx)
174
175     ZEZE2 = diag(2 + gamma_az*Delta_t*as.numeric(a1) + gamma_sz*
      Delta_t*as.numeric(s1), Nx*Nx)
176     ZEZE1 = diag(2 - gamma_az*Delta_t*as.numeric(a1) - gamma_sz*
      Delta_t*as.numeric(s1), Nx*Nx)
177
178     ZAZA2 = diag(2 + omega_az*Delta_t, Nx*Nx)
179     ZAZA1 = diag(2 - omega_az*Delta_t, Nx*Nx)
180
181     ZSZS2 = diag(2 + omega_sz*Delta_t, Nx*Nx)
182     ZSZS1 = diag(2 - omega_sz*Delta_t, Nx*Nx)
183
184     BB2 = diag(2, Nx*Nx)
185     BB1 = diag(2, Nx*Nx)
186
187     MM2 = diag(2, Nx*Nx)
188     MM1 = diag(2, Nx*Nx)
189
190     PP2 = diag(2, Nx*Nx)
191     PP1 = diag(2, Nx*Nx)
192
193     QQ2 = diag(2, Nx*Nx)
194     QQ1 = diag(2, Nx*Nx)
195
196     for(i in 1:(Nx*Nx)){
197
198       i_minus_1=ifelse((i + Nx)%%Nx == 1, i + Nx - 2, i - 1)
199       i_plus_1=ifelse((i + Nx)%%Nx == 0, i - Nx + 2, i + 1)
200       j_minus_1=ifelse(i <= Nx, i + Nx, i - Nx)
201       j_plus_1=ifelse(i > (Nx-1)*Nx, i - Nx, i + Nx)
202       # i_minus_2=ifelse((i + Nx)%%Nx == 1, i + Nx - 3, ifelse((i +
       Nx)%%Nx == 2, i + Nx - 2, i - 2))
203       # i_plus_2=ifelse((i + Nx)%%Nx == 0, i - Nx + 3, ifelse((i +
      Nx)%%Nx == Nx-1, i - Nx + 2, i + 2))
204       # j_minus_2=ifelse(i <= Nx, i, ifelse(i <= 2*Nx, i-Nx, i - 2*
      Nx))
205       # j_plus_2=ifelse(i > (Nx-1)*Nx, i - 2*Nx, ifelse(i > (Nx-2)*
      Nx, i, i + 2*Nx))
```

```
206
207        AA2[i, i_minus_1] = AA2[i, i_minus_1] - r*alpha_a_x
208        AA2[i, j_minus_1] = AA2[i, j_minus_1] - r*alpha_a_y
209        AA2[i, i_plus_1] = AA2[i, i_plus_1] - r*alpha_a_x
210        AA2[i, j_plus_1] = AA2[i, j_plus_1] - r*alpha_a_y
211
212        SS2[i, i_minus_1] = SS2[i, i_minus_1] - r*alpha_s_x
213        SS2[i, j_minus_1] = SS2[i, j_minus_1] - r*alpha_s_y
214        SS2[i, i_plus_1] = SS2[i, i_plus_1] - r*alpha_s_x
215        SS2[i, j_plus_1] = SS2[i, j_plus_1] - r*alpha_s_y
216
217        AA1[i, i_minus_1] = AA1[i, i_minus_1] + r*alpha_a_x
218        AA1[i, j_minus_1] = AA1[i, j_minus_1] + r*alpha_a_y
219        AA1[i, i_plus_1] = AA1[i, i_plus_1] + r*alpha_a_x
220        AA1[i, j_plus_1] = AA1[i, j_plus_1] + r*alpha_a_y
221
222        SS1[i, i_minus_1] = SS1[i, i_minus_1] + r*alpha_s_x
223        SS1[i, j_minus_1] = SS1[i, j_minus_1] + r*alpha_s_y
224        SS1[i, i_plus_1] = SS1[i, i_plus_1] + r*alpha_s_x
225        SS1[i, j_plus_1] = SS1[i, j_plus_1] + r*alpha_s_y
226
227      }
228
229    a2 = solve(AA2) %*% ((AA1 %*% matrix(a1, ncol=1)) + matrix(2*
       Delta_t*omega_aa*za1, ncol=1))
230    s2 = solve(SS2) %*% ((SS1 %*% matrix(s1, ncol=1)) + matrix(2*
       Delta_t*omega_ss*zs1, ncol=1))
231    ze2 = solve(ZEZE2) %*% ((ZEZE1 %*% matrix(ze1, ncol=1)) +
       matrix(2*omega_az*Delta_t*za1, ncol=1) + matrix(2*omega_sz*Delta
       _t*zs1, ncol=1))
232    za2 = solve(ZAZA2) %*% ((ZAZA1 %*% matrix(za1, ncol=1)) +
       matrix(2*Delta_t*gamma_az*a1*ze1, ncol=1))
233    zs2 = solve(ZSZS2) %*% ((ZSZS1 %*% matrix(zs1, ncol=1)) +
       matrix(2*Delta_t*gamma_sz*s1*ze1, ncol=1))
234    b2 = solve(BB2) %*% BB1 %*% matrix(b1, ncol=1)
235    m2 = solve(MM2) %*% MM1 %*% matrix(m1, ncol=1)
236    p2 = solve(PP2) %*% PP1 %*% matrix(p1, ncol=1)
237    q2 = solve(QQ2) %*% QQ1 %*% matrix(q1, ncol=1)
238
239    a2[free_surface]=1
240    s2[free_surface]=1
241
242    a2 -> a1
243    s2 -> s1
244    ze2 -> ze1
245    za2 -> za1
246    zs2 -> zs1
247    b2 -> b1
248    m2 -> m1
249    p2 -> p1
```

```
250       q2 -> q1
251
252       if(as.character(j) %in% as.character(bind_iter)){
253
254         matrix(cbind(a, a1), nrow=Nx*Nx) -> a
255         matrix(cbind(s, s1), nrow=Nx*Nx) -> s
256         matrix(cbind(b, b1), nrow=Nx*Nx) -> b
257         matrix(cbind(m, m1), nrow=Nx*Nx) -> m
258         matrix(cbind(p, p1), nrow=Nx*Nx) -> p
259         matrix(cbind(q, q1), nrow=Nx*Nx) -> q
260         matrix(cbind(ze, ze1), nrow=Nx*Nx) -> ze
261         matrix(cbind(za, za1), nrow=Nx*Nx) -> za
262         matrix(cbind(zs, zs1), nrow=Nx*Nx) -> zs
263
264       }
265
266     }
267
268     # Mass concentration data frame
269     x=seq(0,1,Delta_x)
270     time=seq(0,exposure_time,output_time_step)
271     Nt=length(time)
272
273     data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
            ), Nx)),a) -> df_a
274     names(df_a)=c("x", "Y", paste0("t",time_vals[1:ncol(a)]))# Rename
             columns
275     df_a %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, a=
            value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
             time=as.numeric(time), a=a0*a) %>% arrange(time,Y,x) -> df_a
276
277     # Solvent mass data frame
278     data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
            ), Nx)),s) -> df_s
279     names(df_s)=c("x", "Y", paste0("t",time_vals[1:ncol(s)]))# Rename
             columns
280     df_s %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, s=
            value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
             time=as.numeric(time), s=s0*s) %>% arrange(time,Y,x) -> df_s
281
282     # Empty nanozeolite mass data frame
283     data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
            ), Nx)),ze) -> df_ze
284     names(df_ze)=c("x", "Y", paste0("t",time_vals[1:ncol(ze)]))#
             Rename columns
285     df_ze %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, ze=
            value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
             time=as.numeric(time), ze=z0*ze) %>% arrange(time,Y,x) -> df_ze
286
287     # Analyte-filled nanozeolite mass data frame
```

```r
288    data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
       ), Nx)),za) -> df_za
289    names(df_za)=c("x", "Y", paste0("t",time_vals[1:ncol(za)]))#
       Rename columns
290    df_za %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, za=
       value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
       time=as.numeric(time), za=z0*za) %>% arrange(time,Y,x) -> df_za
291
292    # Solvent-filled nanozeolite mass data frame
293    data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
       ), Nx)),zs) -> df_zs
294    names(df_zs)=c("x", "Y", paste0("t",time_vals[1:ncol(zs)]))#
       Rename columns
295    df_zs %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, zs=
       value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
       time=as.numeric(time), zs=z0*zs) %>% arrange(time,Y,x) -> df_zs
296
297    # Binder data frame
298    data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
       ), Nx)),b) -> df_b
299    names(df_b)=c("x", "Y", paste0("t",time_vals[1:ncol(b)]))# Rename
        columns
300    df_b %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, b=
       value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
       time=as.numeric(time),b=b0*b) %>% arrange(time,Y,x) -> df_b
301
302    # Monomer data frame
303    data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
       ), Nx)),m) -> df_m
304    names(df_m)=c("x", "Y", paste0("t",time_vals[1:ncol(m)]))# Rename
        columns
305    df_m %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, m=
       value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
       time=as.numeric(time),m=m) %>% arrange(time,Y,x) -> df_m
306
307    # Short polymer data frame
308    data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
       ), Nx)),p) -> df_p
309    names(df_p)=c("x", "Y", paste0("t",time_vals[1:ncol(p)]))# Rename
        columns
310    df_p %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, p=
       value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
       time=as.numeric(time),p=p) %>% arrange(time,Y,x) -> df_p
311
312    # Cross-linked polymer data frame
313    data.frame(x=rep(seq(0,1,Delta_x), Nx),Y=sort(rep(seq(0,1,Delta_x
       ), Nx)),q) -> df_q
314    names(df_q)=c("x", "Y", paste0("t",time_vals[1:ncol(q)]))# Rename
        columns
```

```
315    df_q %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, q=
         value) %>% mutate(time=gsub(x=time,pattern="t", replacement=""),
         time=as.numeric(time),q=q) %>% arrange(time,Y,x) -> df_q
316
317    # 3.1 --- Refractive index matrix
318    Vb = b*b0/rhob # cm**3
319    Vm = m*m0/rhom # cm**3
320    Vp = p*m0/rhop # cm**3
321    Vq = q*m0/rhop # cm**3
322    Vze = ze*z0/rhoz # cm**3
323    Vza = za*z0/rhoz # cm**3
324    Vzs = zs*z0/rhoz # cm**3
325    Va = a*a0/rhoa # cm**3
326    Vs = s*s0/rhos # cm**3
327    Vtotal=Vb+Vm+Vp+Vq+Vze+Vza+Vzs+Va+Vs# cm**3
328
329    phi.b=Vb/Vtotal
330    phi.m=Vm/Vtotal
331    phi.p=Vp/Vtotal
332    phi.q=Vq/Vtotal
333    phi.ze=Vze/Vtotal
334    phi.zs=Vzs/Vtotal
335    phi.za=Vza/Vtotal
336    phi.a=Va/Vtotal
337    phi.s=Vs/Vtotal
338
339    Lorentz.Lorenz.RHS = phi.b*(nb*nb - 1)/(nb*nb + 2) + phi.m*(nm*nm
         - 1)/(nm*nm + 2) + phi.p*(np*np - 1)/(np*np + 2) + phi.q*(nq*nq
         - 1)/(nq*nq + 2) + phi.ze*(nze*nze - 1)/(nze*nze + 2) + phi.zs
         *(nzs*nzs - 1)/(nzs*nzs + 2) + phi.za*(nza*nza - 1)/(nza*nza +
         2) + phi.a*(na*na - 1)/(na*na + 2) + phi.s*(ns*ns - 1)/(ns*ns +
         2)
340
341    n=sqrt((2*Lorentz.Lorenz.RHS + 1)/(1 - Lorentz.Lorenz.RHS))
342
343    data.frame(x=rep(seq(0,1,Delta_x), Nx), Y=sort(rep(seq(0,1,Delta_
         x), Nx)), n) -> df_n
344    names(df_n)=c("x","Y", paste0("t",time_vals[1:ncol(n)]))
345    df_n %>% melt(id.vars=c("x","Y")) %>% rename(time=variable, n_
         after_exposure=value) %>% mutate(time=gsub(x=time, pattern="t",
         replacement=""), time=as.numeric(time)) %>% arrange(time,Y,x) ->
          df_n# Additional columns containing info on input
346
347    cbind(df_b,m=df_m$m,p=df_p$p, q=df_q$q, ze=df_ze$ze, za=df_za$za,
          zs=df_zs$zs, a=df_a$a, s=df_s$s, n_after_exposure=df_n$n_after_
         exposure) -> df1
348
349    Nt = length(time)
350    N0=matrix(0, nrow=Nt, ncol=Nx)
351    N1a=matrix(0, nrow=Nt, ncol=Nx)
```

```r
N1b=matrix(0, nrow=Nt, ncol=Nx)
d2=matrix(0, nrow=Nt, ncol=Nx)
int_n2=matrix(0, nrow=Nt, ncol=Nx)
Mean_RI=rep(0,Nt)

for(k in 1:Nt){

  matrix(n[, k], nrow=Nx) -> n1

  n2=n1**2

  Mean_RI[k]=Delta_x*Delta_x/4*(n1[1,1] + n1[Nx,1] + n1[1,Nx] +
  n1[Nx,Nx] + sum(2*n1[1,(2:(Nx-1))]) + sum(2*n1[(2:(Nx-1)),1]) +
  sum(2*n1[Nx,(2:(Nx-1))]) + sum(2*n1[(2:(Nx-1)),Nx]) + sum(4*n1
  [(2:(Nx-1)),(2:(Nx-1))]) )

  for(i in 1:Nx){

    n1[, i] -> n1i
    N0[k, i]=Delta_x/3*(n1i[1] + sum(2*n1i[times_2]) + sum(4*n1i[
  times_4]) + n1i[Nx])

    n1[, i]*cos(2*pi*x) -> n1cos
    N1a[k, i]=2*Delta_x/3*(n1cos[1] + sum(2*n1cos[times_2]) + sum
  (4*n1cos[times_4]) + n1cos[Nx])

    n1[, i]*sin(2*pi*x) -> n1sin
    N1b[k, i]=2*Delta_x/3*(n1sin[1] + sum(2*n1sin[times_2]) + sum
  (4*n1sin[times_4]) + n1sin[Nx])

  }


  n.Fourier=matrix(0, nrow=Nx, ncol=Nx)

  for(i in 1:Nx){

    n.Fourier[,i] = N0[k, i] + N1a[k, i]*cos(2*pi*x) + N1b[k, i]*
  sin(2*pi*x)

  }

  diff = abs(n1 - n.Fourier)**2

  for(i in 1:Nx){
    d2[k,i] = Delta_x/3*(diff[1, i] + sum(2*diff[times_2, i]) +
  sum(4*diff[times_4, i]) + diff[Nx, i])
    int_n2[k,i] = Delta_x/3*(n2[1, i] + sum(2*n2[times_2, i]) +
  sum(4*n2[times_4, i]) + n2[Nx, i])
  }
```

```r
393
394    }# End for loop
395
396    delta=d2/int_n2
397    Delta_n=2*sqrt(N1a*N1a + N1b*N1b)
398
399    data.frame(
400      time=rep(time, Nx),
401      Y=sort(rep(seq(0,1,Delta_x), Nt)),
402      Delta_n=melt(Delta_n)$value,
403      delta=melt(delta)$value
404    ) -> df2
405
406    df1 %>% mutate(
407
408      Delta_n=0,
409      delta=0,
410      Mean_RI=0
411
412    ) %>% arrange(time,Y,x) -> df1
413
414    for(i in time){
415      for(j in seq(0,1,Delta_x)){
416        df1[df1$time==i & df1$Y==j, "Delta_n"] = df2[df2$time==i &
      df2$Y==j, "Delta_n"]
417        df1[df1$time==i & df1$Y==j, "delta"] = df2[df2$time==i & df2$
      Y==j, "delta"]
418      }
419    }
420
421    for(i in 1:length(time)){
422      df1[df1$time==time[i], "Mean_RI"]=Mean_RI[i]
423    }
424
425    df1 %>% mutate(
426      theta_B = asin(lambda_probe/2/Mean_RI/Lambda_1) - phi_r_1,
427      Delta_theta_B = theta_B1 - theta_B,
428      Delta_phi_r = 0
429    ) -> df1
430
431    if(df0 %>% subset(Y==0) %>% distinct(Klein.Cook) %>% pull(Klein.
      Cook) < 10){
432
433      df1 %>% mutate(nu=pi*Delta_n*T_1/lambda_probe, xi=0, J0=nu/2,
      J1=J0) -> df1
434
435      for(l in 1:100){
436
437        df1$J1 = df1$J1 + ((-1)**l)/factorial(l)/factorial(l+1)*df1$
      J0**(2*l + 1)
```

```r
438
439        }
440
441    df1 %>% mutate(
442
443        eta=J1*J1,
444        Geometry="Planar"
445
446    ) %>% select(-J1,-J0) -> df1
447
448  } else {
449
450        df1 %>% mutate(
451
452          nu=pi*Delta_n*T_1/lambda_probe/cos(theta_B),
453          xi=pi*T_1*Delta_phi_r/Lambda_1,
454          eta=sin(sqrt(xi*xi + nu*nu))**2/sqrt(1 + (xi*xi)/(nu*nu)),
455          Geometry="Volume"
456
457        ) -> df1
458
459  }
460
461  df1 %>% subset(time==0) %>% rename(pre_exposure_eta=eta) %>%
        distinct(Y,pre_exposure_eta) -> df4
462
463  df1 %>% mutate(pre_exposure_eta=0) -> df1
464
465  for(i in seq(0,1,Delta_x)){
466    df1[df1$Y==i, "pre_exposure_eta"] = df4[df4$Y==i,"pre_exposure_
      eta"]
467  }
468
469  df1 %>% mutate(
470
471    normalized_eta=eta/pre_exposure_eta,
472    lambda_r_t=2*Mean_RI*Lambda_1*sin(theta_B),
473    nze=nze,
474    nza=nza,
475    nzs=nzs,
476    na=na,
477    ns=ns,
478    z0=z0,
479    a0=a0,
480    s0=s0,
481    b0=b0,
482    tau_c_s=tau_c_s,
483    tau_c_a=tau_c_a,
484    tau_e_s=tau_e_s,
485    tau_e_a=tau_e_a,
```

```
486       tau_y_s=T_1*T_1/Ds,
487       tau_y_a=T_1*T_1/Da,
488       tau_x_s=x_hat*x_hat/Ds,
489       tau_x_a=x_hat*x_hat/Da,
490       gamma_a=gamma_a,
491       gamma_s=gamma_s,
492       omega_a=omega_a,
493       omega_s=omega_s,
494       rhoz=rhoz,
495       rhoa=rhoa,
496       rhos=rhos,
497       Ds=Ds,
498       Da=Da,
499       wt_pc=wt_pc,
500       lambda_probe=lambda_probe,
501       T_1=T_1,
502       pre_exposure_Bragg_angle=theta_B1,
503       lpmm=lpmm,
504       exposure_time=exposure_time,
505       output_time_step=output_time_step,
506       model="sensor_2D_model_v2"
507     ) %>% arrange(time,Y,x) -> df1
508
509     return(df1)
510
511 }
```

Appendix C

R Code: Numerical Simulations & Plot Results

Chapter 1

Fig. 1.5

```r
data.frame() -> ch1.df1

for(k in c(1e2,5e2,1e3,2e3)){
  coupled_cross_diffusion_v5(end_exp=50, total.time=50, lpmm=k, xi
    =0, Dmz=0, Dpz=0, Dzp=0, Dzq=0, wt.pc=0, F0=0.1*1**0.3, Gamma=0,
     Dp=0) %>% distinct(x,time,Delta.n,n,lpmm) %>% rbind(ch1.df1) ->
     ch1.df1
}

ggplot(data=subset(ch1.df1, time==50), mapping=aes(x=x, y=n, color=
    factor(lpmm))) + geom_line(size=3)

ggplot(data=ch1.df1, mapping=aes(x=time, y=Delta.n, color=factor(
    lpmm))) + geom_point(size=4) + geom_line(size=1)
```

Fig. 1.7

```r
data.frame() -> ch1.df2

for(k in c(1e2,5e2,1e3,2e3)){
  coupled_cross_diffusion_v5(end_exp=50, total.time=50, lpmm=k, xi
    =0, Dmz=0, Dpz=0, Dzp=0, Dzq=0, wt.pc=0, F0=0.1*1**0.3, Gamma=1)
     %>% distinct(time,Delta.n,distortion,lpmm) %>% rbind(ch1.df2)
    -> ch1.df2
}

ggplot(data=ch1.df2, mapping=aes(x=time, y=Delta.n, color=factor(
    lpmm))) + geom_point(size=4) + geom_line(size=1)
```

```
8
9 ggplot(data=ch1.df2, mapping=aes(x=time, y=distortion, color=factor
     (lpmm))) + geom_point(size=4) + geom_line(size=1)
```

Fig. 1.8

```
1 data.frame() -> ch1.df3
2
3 for(k in c(10,1e2,1e3)){
4   coupled_cross_diffusion_v5(end_exp=50, total.time=50, lpmm=k, xi
     =0, Dmz=0, Dpz=0, Dzp=0, Dzq=0, wt.pc=0, F0=0.1*4.8**0.3, Gamma
     =1) %>% mutate(Lambda=1/10/lpmm, alpha=Dm/Lambda/Lambda, beta=F0
     , kappa=alpha/beta) %>% distinct(time,Delta.n,distortion,kappa)
     %>% rbind(ch1.df3) -> ch1.df3
5 }
6
7 ggplot(data=ch1.df3, mapping=aes(x=time, y=Delta.n, color=factor(
     kappa))
8 ) + geom_point(data=subset(ch1.df3, time %in% seq(0,50,5)),size=4)
     + geom_line(size=1)
9
10 ggplot(data=ch1.df3, mapping=aes(x=time, y=distortion, color=factor
     (kappa))
11 ) + geom_point(data=subset(ch1.df3, time %in% seq(0,50,5)),size=4)
     + geom_line(size=1)
```

Fig. 1.9

```
1 data.frame() -> ch1.df4
2
3 for(k in c(1,5,10,0.1)){
4   coupled_cross_diffusion_v5(end_exp=50/k, total.time=50, lpmm
     =1000, xi=0, Dmz=0, Dpz=0, Dzp=0, Dzq=0, wt.pc=0, F0=0.1*k**0.3,
      Gamma=1) %>% mutate(F0=0.1*k**0.3) %>% distinct(time,Delta.n,
     distortion,I0) %>% rbind(ch1.df4) -> ch1.df4
5 }
6
7 ggplot(data=ch1.df4, mapping=aes(x=time, y=Delta.n, color=factor(I0
     ))) + geom_point(size=4) + geom_line(size=1)
8
9 ggplot(data=ch1.df4, mapping=aes(x=time, y=distortion, color=factor
     (I0))) + geom_point(size=4) + geom_line(size=1)
```

Fig. 1.10

```
1 data.frame() -> ch1.df5
2
3 for(k in c(1.4,1.5,1.6,1.7)){
```

```
4    coupled_cross_diffusion_v5(end_exp=50, total.time=50, lpmm=1000,
       xi=0, Dmz=0, Dpz=0, Dzp=0, Dzq=0, wt.pc=0, F0=0.1*1**0.3, Gamma
       =1, nb=k) %>% distinct(time,Delta.n,distortion,nb) %>% rbind(ch1
       .df5) -> ch1.df5
5  }
6
7  ggplot(data=ch1.df5, mapping=aes(x=time, y=Delta.n, color=factor(nb
       ))) +geom_point(size=4) +geom_line(size=1)
8
9  ggplot(data=ch1.df5, mapping=aes(x=time, y=distortion, color=factor
       (nb))) + geom_point(size=4) +geom_line(size=1)
```

Chapter 2

Fig. 2.3

```
1  data.frame() -> ch2.df2
2
3  for(k in c(0,2,4,6)){
4    coupled_cross_diffusion_v5(total.time=100, end_exp=90, F0
       =0.1*0.5**0.3, Gamma=0.6, rhoz=2.24, nz=1.456, xi=0, epsilon.mz
       =0, epsilon.pz=k, epsilon.qz=0) %>% distinct(x,mass.q,mass.z,
       redistributed,time,Delta.n,epsilon.pz) %>% rbind(ch2.df2) -> ch2
       .df2
5  }; rm(k)
6
7  ggplot(data=ch2.df2 %>% subset(time==1e2), mapping=aes(x=x, y=mass.
       q, color=factor(epsilon.pz))) + geom_line(size=3)
8
9  ggplot(data=ch2.df2 %>% subset(time==1e2), mapping=aes(x=x, y=mass.
       z, color=factor(epsilon.pz))) + geom_line(size=3)
10
11 ggplot(data=ch2.df2, mapping=aes(x=time, y=1e2*redistributed, color
       =factor(epsilon.pz))) + geom_line(size=3)
12
13 ggplot(data=ch2.df2, mapping=aes(x=time, y=1e3*Delta.n, color=
       factor(epsilon.pz))) + geom_line(size=3)
```

Fig. 2.4

```
1  data.frame() -> ch2.df3
2
3  for(k in c(0,2,4,6)){
4    coupled_cross_diffusion_v5(total.time=100, end_exp=90, F0
       =0.1*0.5**0.3, Gamma=0.6, rhoz=2.24, nz=1.456, xi=0, epsilon.mz
       =0, epsilon.qz=k, epsilon.pz=0) %>% distinct(x,mass.q,mass.z,
       time,Delta.n,epsilon.qz,redistributed) %>% rbind(ch2.df3) -> ch2
       .df3
```

```
5  }; rm(k)
6
7  ggplot(data=ch2.df3 %>% subset(time==1e2), mapping=aes(x=x, y=mass.
       q, color=factor(epsilon.qz))) + geom_line(size=3)
8
9  ggplot(data=ch2.df3 %>% subset(time==1e2), mapping=aes(x=x, y=mass.
       z, color=factor(epsilon.qz))) + geom_line(size=3)
10
11 ggplot(data=ch2.df3, mapping=aes(x=time, y=1e2*redistributed, color
       =factor(epsilon.qz))) + geom_line(size=3)
12
13 ggplot(data=ch2.df3, mapping=aes(x=time, y=1e3*Delta.n, color=
       factor(epsilon.qz))) + geom_line(size=3)
```

Fig. 2.5

```
1  data.frame() -> ch2.df4
2
3  for(k in c(0,2,4,6)){
4    coupled_cross_diffusion_v5(total.time=100, end_exp=90, F0
       =0.1*0.5**0.3, Gamma=0.6, rhoz=2.24, nz=1.456, xi=0, epsilon.mz
       =0, epsilon.qz=k, epsilon.pz=k) %>% distinct(x,mass.q,mass.z,n,
       time,Delta.n,epsilon.qz,redistributed) %>% rbind(ch2.df4) -> ch2
       .df4
5  }; rm(k)
6
7  ggplot(data=ch2.df4 %>% subset(time==1e2), mapping=aes(x=x, y=mass.
       q, color=factor(epsilon.qz))) + geom_line(size=3)
8
9  ggplot(data=ch2.df4 %>% subset(time==1e2), mapping=aes(x=x, y=mass.
       z, color=factor(epsilon.qz))) + geom_line(size=3)
10
11 ggplot(data=ch2.df4, mapping=aes(x=time, y=1e2*redistributed, color
       =factor(epsilon.qz))) + geom_line(size=3)
12
13 ggplot(data=ch2.df4, mapping=aes(x=time, y=1e3*Delta.n, color=
       factor(epsilon.qz))) + geom_line(size=3)
```

Fig. 2.6

```
1  data.frame() -> ch2.df5
2
3  for(k in c(0,0.2,0.4,0.6)){
4    coupled_cross_diffusion_v5(total.time=100, end_exp=90, F0
       =0.1*0.5**0.3, Gamma=0.6, rhoz=2.24, nz=1.456, xi=k, epsilon.mz
       =0, epsilon.qz=6, epsilon.pz=6) %>% distinct(x,mass.q,mass.z,n,
       time,Delta.n,xi,redistributed) %>% rbind(ch2.df5) -> ch2.df5
5  }; rm(k)
6
```

```r
7  ggplot(data=ch2.df5 %>% subset(time==1e2), mapping=aes(x=x, y=mass.
       q, color=factor(xi))) + geom_line(size=3)
8
9  ggplot(data=ch2.df5 %>% subset(time==1e2), mapping=aes(x=x, y=mass.
       z, color=factor(xi))) + geom_line(size=3)
10
11 ggplot(data=ch2.df5, mapping=aes(x=time, y=1e2*redistributed, color
       =factor(xi))) + geom_line(size=3)
12
13 ggplot(data=ch2.df5, mapping=aes(x=time, y=1e3*Delta.n, color=
       factor(xi))) + geom_line(size=3)
```

Fig. 2.7

```r
1  data.frame() -> ch2.df38a
2
3  for(j in c(0,0.8,1.6,3.2)){
4    for(k in seq(0,10e-2,0.5e-2)){
5      coupled_cross_diffusion_v5(F0=0.1*0.5**0.3, Gamma=0.6, rhoz
       =2.24, nz=1.456, wt.pc=k, xi=j, Dmz=0, Dpz=13*6.35e-10, Dzp=13*1
       e-10, Dzq=13*1e-10) %>% subset(time==max(time)) %>% distinct(wt.
       pc,Delta.n,xi) %>% rbind(ch2.df38a) -> ch2.df38a
6    }
7  }; rm(j,k)
8
9  data.frame() -> ch2.df38b
10
11 for(i in c(0,11,13,15)){
12   for(k in seq(0,10e-2,0.5e-2)){
13     coupled_cross_diffusion_v5(F0=0.1*0.5**0.3, Gamma=0.6, rhoz
       =2.24, nz=1.456, wt.pc=k, xi=1.6, Dmz=0, Dpz=i*6.35e-10, Dzp=i*1
       e-10, Dzq=i*1e-10) %>% subset(time==max(time)) %>% distinct(wt.
       pc,Delta.n,Dzq,Dz) %>% rbind(ch2.df38b) -> ch2.df38b
14   }
15 }; rm(i,k)
16
17 ggplot(data=ch2.df38a, mapping=aes(x=1e2*wt.pc, y=1e3*Delta.n,
       shape=factor(xi.z))) + geom_line(size=1) + geom_point(size=3)
18
19 ggplot(data=ch2.df38b %>% mutate(epsilon.z=Dzq/Dz), mapping=aes(x=1
       e2*wt.pc, y=1e3*Delta.n, shape=factor(epsilon.z))) + geom_line(
       size=1) + geom_point(size=3)
```

Fig. 2.8

```r
1  data.frame(time=as.numeric(), Delta.n=as.numeric(), wt.pc=
       as.numeric(), x=as.numeric(), density.m=as.numeric(), density.p=
       as.numeric(), density.q=as.numeric(), density.z=as.numeric()) ->
        ch2.df39
```

```
2
3 for(k in c(0,1e-2,2e-2,5e-2)){
4   coupled_cross_diffusion_v5(F0=0.1*0.5**0.3, Gamma=0.6, rhoz=2.24,
      nz=1.456, wt.pc=k, xi=1.6, Dmz=0, Dpz=13*6.35e-10, Dzp=13*1e
      -10, Dzq=13*1e-10) %>% distinct(time,Delta.n,wt.pc) %>% rbind(
      ch2.df39) -> ch2.df39
5 }; rm(k)
6
7 ggplot(data=ch2.df39, mapping=aes(x=time, y=1e3*Delta.n, shape=
      factor(1e2*wt.pc))) + geom_line(size=1) + geom_point(size=4)
```

Fig. 2.9

```
1 ggplot(data=ch2.df39 %>% subset(time==100), mapping=aes(x=x, y=mass
      .m, shape=factor(1e2*wt.pc))) + geom_line(size=1) + geom_point(
      size=4)
2
3 ggplot(data=ch2.df39 %>% subset(time==100), mapping=aes(x=x, y=mass
      .p, shape=factor(1e2*wt.pc))) + geom_line(size=1) + geom_point(
      size=4)
4
5 ggplot(data=ch2.df39 %>% subset(time==100), mapping=aes(x=x, y=mass
      .q, shape=factor(1e2*wt.pc))) + geom_line(size=1) + geom_point(
      size=4)
6
7 ggplot(data=ch2.df39 %>% subset(time==100), mapping=aes(x=x, y=mass
      .z, shape=factor(1e2*wt.pc))) + geom_line(size=1) + geom_point(
      size=4)
```

Fig. 2.10

```
1 data.frame() -> ch2.df40a
2
3 for(k in c(400,600,800,1000)){
4   for(i in seq(0,10e-2,5e-3)){
5     coupled_cross_diffusion_v5(lpmm=k, xi=1.6, Dmz=0, Dpz=13*6.35e
      -10, Dzp=13*1e-10, Dzq=13*1e-10, wt.pc=i, F0=0.1*0.5**0.3, Gamma
      =0.6, rhoz=2.24, nz=1.456) %>% subset(time==max(time)) %>%
      distinct(wt.pc,Delta.n,distortion,lpmm) %>% rbind(ch2.df40a) ->
      ch2.df40a
6   }
7 }
8
9 ggplot(data=ch2.df40a, mapping=aes(x=1e2*wt.pc, y=1e2*Delta.n,
      shape=factor(lpmm))) + geom_line(size=1) + geom_point(size=4)
10
11 ggplot(data=ch2.df40a, mapping=aes(x=1e2*wt.pc, y=1e5*distortion,
      shape=factor(lpmm))) + geom_line(size=1) + geom_point(size=4)
```

Fig. 2.11

```
1  data.frame() -> ch2.df40b
2
3  for(k in c(0.5,5,25,100)){
4    for(i in seq(0,10e-2,5e-3)){
5      coupled_cross_diffusion_v5(lpmm=1e3, xi=1.6, Dmz=0, Dpz=13*6.35
       e-10, Dzp=13*1e-10, Dzq=13*1e-10, wt.pc=i, F0=0.1*k**0.3, Gamma
       =0.6, rhoz=2.24, nz=1.456) %>% subset(time==max(time)) %>%
       mutate(F0=0.1*k**0.3) %>% distinct(wt.pc,Delta.n,distortion,I0)
       %>% rbind(ch2.df40b) -> ch2.df40b
6    }
7  }
8
9  ggplot(data=ch2.df40b, mapping=aes(x=1e2*wt.pc, y=1e2*Delta.n,
       shape=factor(I0))) + geom_line(size=1) + geom_point(size=4)
10
11 ggplot(data=ch2.df40b, mapping=aes(x=1e2*wt.pc, y=1e5*distortion,
       shape=factor(I0))) + geom_line(size=1) + geom_point(size=4)
```

Fig. 2.12

```
1  data.frame() -> ch2.df41
2
3  for(i in seq(0,10e-2,0.5e-2)){
4    for(j in seq(-4e-2,+4e-2,5e-3)){
5      coupled_cross_diffusion_v5(Dpz=13*6.35e-10, Dzq=13*1e-10, Dzp
       =13*1e-10, xi=1.5, nz=1.5+j, wt.pc=i, lpmm=1e2) %>% subset(time
       ==max(time)) %>% distinct(nz,wt.pc,Delta.n) %>% rbind(ch2.df41)
       -> ch2.df41
6    }
7  }; rm(i,j)
8
9  ch2.df41 %>% pull(nz) %>% unique %>% sort -> nz.vals
10 ch2.df41 %>% pull(wt.pc) %>% unique %>% sort -> doping.vals
11
12 heatmap_matrix=data.frame(wt.pc=doping.vals)
13
14 for(j in nz.vals){
15
16   arrange(subset(ch2.df41, nz==j), wt.pc) %>% select(nz,Delta.n) ->
       temp
17
18   temp[, as.character(j)] = temp$Delta.n
19
20   temp %>% select(-Delta.n,-nz) -> temp
21
22   heatmap_matrix=cbind(heatmap_matrix,temp)
23
24   rm(temp)
```

```
25
26  }; rm(j)
27
28  row.names(heatmap_matrix)=format(doping.vals)
29  heatmap_matrix %>% select(-wt.pc) -> heatmap_matrix
30  as.matrix(heatmap_matrix) -> heatmap_matrix
31
32  filled.contour(
33    y=as.numeric(nz.vals),
34    x=1e2*as.numeric(doping.vals),
35    z=heatmap_matrix,
36    nlevels=20,
37    main=bquote(Delta*n),
38    color.palette = function(n) hcl.colors(n, "RdBu"),
39    plot.axes = {
40      axis(1, seq(0,10,1))
41      axis(2, seq(0,2.1,0.1))
42      contour(
43        y=seq(1.3,2.1,length=ncol(heatmap_matrix)),
44        x=seq(0,10,length=nrow(heatmap_matrix)),
45        z=heatmap_matrix,
46        nlevels=20,
47        add = TRUE,
48        lwd = 2
49      )
50    }
51  )
```

Fig. 2.13

```
1  ggplot(data=ch2.df41 %>% subset(nz %in% c(1.46,1.82)), mapping=aes(
       x=1e2*wt.pc, y=1e3*Delta.n, shape=factor(1.64-nz))) + geom_line(
       size=1) + geom_point(size=4)
2
3  ggplot(data=ch2.df41 %>% subset(nz %in% c(1.46,1.82)), mapping=aes(
       x=1e2*wt.pc, y=1e9*distortion, shape=factor(1.64-nz))) +
       geom_line(size=1) + geom_point(size=4)
```

Chapter 3

Fig. 3.1

```
1  data.frame(lpmm=as.numeric(), shrinkage=as.numeric(), I0=as.numeric
       ()) -> df1
2
3  for(i in seq(5e2,2e3,1e2)){
4    for(j in c(1,5,10)){
```

```
5    coupled_cross_diffusion_v7(total.time=80, end_exp=80/j, I0=0.1*
     j**0.3, lpmm=i, wt.pc=0) %>% subset(time==max(time)) %>%
     distinct(lpmm,shrinkage,I0) %>% rbind(df1) -> df1
6  }
7 }; rm(i,j)
8
9 ggplot(data=df1, mapping=aes(x=lpmm, y=1e2*shrinkage, color=factor(
     I0))) + geom_line(size=1) + geom_point(size=4)
```

Fig. 3.2

```
1 data.frame() -> df2
2
3 data.frame() -> df3
4
5 for(i in seq(1,6,0.5)){
6   for(j in c(1,5,10)){
7     coupled_cross_diffusion_v7(total.time=80, end_exp=80/j, I0=0.1*
     j**0.3, wt.pc=0, b0=i) %>% subset(time==max(time)) %>% mutate(F0
     =0.1*j**0.3, b0=mass.b) %>% distinct(b0,shrinkage,I0) %>% rbind(
     df2) -> df2
8   }
9 }; rm(i,j)
10
11 for(i in seq(0.05,1.15,0.05)){
12   for(j in c(1,5,10)){
13     coupled_cross_diffusion_v7(total.time=80, end_exp=80/j, I0=0.1*
     j**0.3, wt.pc=0, rhop=1.15+i) %>% subset(time==max(time)) %>%
     mutate(F0=0.1*j**0.3) %>% distinct(rhop,rhom,shrinkage,I0) %>%
     rbind(df3) -> df3
14   }
15 }; rm(i,j)
16
17 ggplot(data=df2, mapping=aes(x=b0, y=1e2*shrinkage, color=factor(I0
     ))) + geom_point(size=4) + geom_smooth(formula = y ~ poly(x,
     degree=2), size=1, se=FALSE, method=lm)
18
19 ggplot(data=df3, mapping=aes(x=rhop-rhom, y=1e2*shrinkage, color=
     factor(I0))) + geom_point(size=4) + geom_smooth(formula = y ~
     poly(x, degree=2), size=1, se=FALSE, method=lm)
```

Fig. 3.3

```
1 data.frame(wt.pc=as.numeric(), shrinkage=as.numeric()) -> df4
2
3 for(i in c(0,1e-2,2.5e-2,5e-2,7.5e-2)){
4   coupled_cross_diffusion_v7(total.time=80, end_exp=80, I0
     =0.1*1**0.3, wt.pc=i) %>% subset(time==max(time)) %>% distinct(
     wt.pc,shrinkage) %>% rbind(df4) -> df4
```

```
5 }; rm(i)
6
7 ggplot(data=df4, mapping=aes(x=1e2*wt.pc, y=1e2*shrinkage)) +
     geom_line(size=1) + geom_point(size=4)
```

Fig. 3.8

```
1 slanted_grating_simulation_v21(slant.angle = 10, total.time=80, I0
     =0.1*1**0.3) -> df5
2
3 ggplot(data=subset(df5, time %in% c(0,10,80) & y==0), mapping=aes(x
     =x, y=mass.m, color=factor(time))) + geom_line(size=3)
4
5 ggplot(data=subset(df5, time %in% c(0,10,80) & y==1), mapping=aes(x
     =x, y=mass.m, color=factor(time))) + geom_line(size=3)
```

Fig. 3.9

```
1 ggplot(data=subset(df5, time %in% c(0,10,80) & y==0), mapping=aes(x
     =x, y=mass.p, color=factor(time))) + geom_line(size=3)
2
3 ggplot(data=subset(df5, time %in% c(0,10,80) & y==1), mapping=aes(x
     =x, y=mass.p, color=factor(time))) + geom_line(size=3)
```

Fig. 3.10

```
1 ggplot(data=subset(df5, time %in% c(0,10,80) & y==0), mapping=aes(x
     =x, y=mass.q, color=factor(time))) + geom_line(size=3)
2
3 ggplot(data=subset(df5, time %in% c(0,10,80) & y==1), mapping=aes(x
     =x, y=mass.q, color=factor(time))) + geom_line(size=3)
```

Fig. 3.11

```
1 ggplot(data=subset(df5, time %in% c(0,10,80) & y==0), mapping=aes(x
     =x, y=mass.z, color=factor(time))) + geom_line(size=3)
2
3 ggplot(data=subset(df5, time %in% c(0,10,80) & y==1), mapping=aes(x
     =x, y=mass.z, color=factor(time))) + geom_line(size=3)
```

Fig. 3.12

```
1 data.frame() -> df6
2
3 for(i in c(0,0.05,0.1,0.15)){
```

```
 4   slanted_grating_simulation_v21(total.time = 50, I0=1, rhom=1.15,
       rhop=1.15+i, rhoq=1.15+i) %>%
 5       distinct(time,Delta_phi_r,rhop,rhom,b0,zeta,I0) %>%
 6       rbind(df6) -> df6
 7 }
 8
 9 for(i in seq(3,6,1)){
10   slanted_grating_simulation_v21(total.time = 50, I0=1, b0=i) %>%
11       distinct(time,Delta_phi_r,rhop,rhom,b0,zeta,I0) %>%
12       rbind(df6) -> df6
13 }
14
15 for(i in c(0,139,277)){
16   slanted_grating_simulation_v21(total.time = 50, I0=1, zeta=i) %>%
17       distinct(time,Delta_phi_r,rhop,rhom,b0,zeta,I0) %>%
18       rbind(df6) -> df6
19 }
20
21 for(i in c(1,5,10)){
22   slanted_grating_simulation_v21(total.time = 50/i, I0=i) %>%
23       distinct(time,Delta_phi_r,rhop,rhom,b0,zeta,I0) %>%
24       rbind(df6) -> df6
25 }; rm(i)
26
27 ggplot(df6 %>% subset(b0==5 & zeta==139 & I0==1), aes(x=time, y=
       Delta_phi_r, color=factor(rhop-rhom))) + geom_line(size=2)
28
29 ggplot(df6 %>% subset(rhop==1.3 & zeta==139 & I0==1), aes(x=time, y
       =Delta_phi_r, color=factor(b0))) + geom_line(size=2)
30
31 ggplot(df6 %>% subset(b0==5 & rhop==1.3 & I0==1), aes(x=time, y=
       Delta_phi_r, color=factor(zeta))) + geom_line(size=2)
32
33 ggplot(df6 %>% subset(b0==5 & zeta==139 & rhop==1.3), aes(x=time, y
       =Delta_phi_r, color=factor(I0))) + geom_line(size=2)
```

Fig. 3.13

```
1 ggplot(data=subset(df5, time %in% c(0,10,80) & y==0), mapping=aes(x
    =x, y=n, color=factor(time))) + geom_line(size=3)
2
3 ggplot(data=subset(df5, time %in% c(0,10,80) & y==1), mapping=aes(x
    =x, y=n, color=factor(time))) + geom_line(size=3)
```

Fig. 3.14

```
1 data.frame() -> df7
2
3 for(i in c(0,1e-2,2e-2,5e-2)){
```

```r
4    slanted_grating_simulation_v21(total.time = 50, I0=1, wt.pc=i, nz
       =1.366) %>%
5      distinct(time,mean_RI,wt.pc,nz) %>%
6      rbind(df7) -> df7
7  }; rm(i)

8

9  for(i in c(1.456,1.6,2.1)){
10   slanted_grating_simulation_v21(total.time = 50, I0=1, wt.pc=5e-2,
       nz=i) %>%
11     distinct(time,mean_RI,wt.pc,nz) %>%
12     rbind(df7) -> df7
13 }; rm(i)

14

15 ggplot(df7 %>% subset(nz==1.366), aes(x=time, y=mean_RI, color=
      factor(1e2*wt.pc))) + geom_line(size=2)

16

17 ggplot(df7 %>% subset(wt.pc==5e-2), aes(x=time, y=mean_RI, color=
      factor(nz))) + geom_line(size=2)
```

Fig. 3.15

```r
1  data.frame() -> df1

2

3  for(i in seq(0,5e-2,5e-3)){
4    slanted_grating_simulation_v21(wt.pc=i, I0=1, slant.angle = 10)
       %>%
5      subset(time==max(time)) %>%
6      distinct(wt.pc,Actual.Shrinkage,Apparent.Shrinkage) %>%
7      rbind(df1) -> df1
8  }

9

10 ggplot(df1 %>% melt(id.vars="wt.pc"), aes(x=1e2*wt.pc, y=1e2*value,
       color=factor(variable))) + geom_point(size=4) + geom_line(size
      =1)

11

12 data.frame() -> df2

13

14 for(i in seq(1.3,2.1,.1)){
15   slanted_grating_simulation_v21(I0=1, slant.angle = 10, nz=i) %>%
16     subset(time==max(time)) %>%
17     distinct(nz,Actual.Shrinkage,Apparent.Shrinkage) %>%
18     rbind(df2) -> df2
19 }

20

21 ggplot(df2 %>% melt(id.vars="nz"), aes(x=nz, y=1e2*value, color=
      factor(variable))) + geom_point(size=4) + geom_line(size=1)

22

23 data.frame() -> df3

24

25 for(i in seq(1.5,2.5,.1)){
```

```
26    slanted_grating_simulation_v21(I0=1, slant.angle = 10, rhoz=i)
      %>%
27      subset(time==max(time)) %>%
28      distinct(rhoz,Actual.Shrinkage,Apparent.Shrinkage) %>%
29      rbind(df3) -> df3
30 }
31
32 ggplot(df3 %>% melt(id.vars="rhoz"), aes(x=rhoz, y=1e2*value, color
      =factor(variable))) + geom_point(size=4) + geom_line(size=1)
```

Fig. 3.16

```
1 data.frame() -> df1
2
3 for(i in seq(1.5,1.6,.02)){
4   slanted_grating_simulation_v21(nb=i, I0=1, slant.angle = 10) %>%
5     subset(time==max(time)) %>%
6     distinct(nb,Actual.Shrinkage,Apparent.Shrinkage) %>%
7     rbind(df1) -> df1
8 }
9
10 ggplot(df1 %>% melt(id.vars="nb"), aes(x=nb, y=1e2*value, color=
      factor(variable))) + geom_point(size=4) + geom_line(size=1)
11
12 data.frame() -> df2
13
14 for(i in seq(3,6,.5)){
15   slanted_grating_simulation_v21(I0=1, slant.angle = 10, b0=i) %>%
16     subset(time==max(time)) %>%
17     distinct(b0,Actual.Shrinkage,Apparent.Shrinkage) %>%
18     rbind(df2) -> df2
19 }
20
21 ggplot(df2 %>% melt(id.vars="b0"), aes(x=b0, y=1e2*value, color=
      factor(variable))) + geom_point(size=4) + geom_line(size=1)
```

Fig. 3.17

```
1 data.frame() -> df1
2
3 for(i in seq(0,.2,.02)){
4   slanted_grating_simulation_v21(rhom=1.15, rhop=1.15+i, rhoq=1.15+
      i, I0=1, slant.angle = 10) %>%
5     subset(time==max(time)) %>%
6     distinct(rhop,Actual.Shrinkage,Apparent.Shrinkage) %>%
7     rbind(df1) -> df1
8 }
9
```

```r
10  ggplot(df1 %>% melt(id.vars="rhop"), aes(x=rhop-1.15, y=1e2*value,
        color=factor(variable))) + geom_point(size=4) + geom_line(size
        =1)
11
12  data.frame() -> df2
13
14  for(i in seq(0,.15,.03)){
15    slanted_grating_simulation_v21(I0=1, slant.angle = 10, nm=1.55,
        np=1.55+i, nq=1.55+i) %>%
16      subset(time==max(time)) %>%
17      distinct(np,Actual.Shrinkage,Apparent.Shrinkage) %>%
18      rbind(df2) -> df2
19  }
20
21  ggplot(df2 %>% melt(id.vars="np"), aes(x=np-1.55, y=1e2*value,
        color=factor(variable))) + geom_point(size=4) + geom_line(size
        =1)
```

Fig. 3.18

```r
1   data.frame() -> df1
2
3   for(i in seq(250,2e3,250)){
4     slanted_grating_simulation_v21(lpmm=i, I0=1, slant.angle = 10)
        %>%
5       subset(time==max(time)) %>%
6       distinct(lpmm,Actual.Shrinkage,Apparent.Shrinkage) %>%
7       rbind(df1) -> df1
8   }
9
10  ggplot(df1 %>% melt(id.vars="lpmm"), aes(x=lpmm, y=1e2*value, color
        =factor(variable))) + geom_point(size=4) + geom_line(size=1)
11
12  data.frame() -> df2
13
14  for(i in seq(1,10,1)){
15    slanted_grating_simulation_v21(total.time=1e2, end_exp=1e2/i, I0=
        i, slant.angle = 10) %>%
16      subset(time==max(time)) %>%
17      distinct(I0,Actual.Shrinkage,Apparent.Shrinkage) %>%
18      rbind(df2) -> df2
19  }
20
21  ggplot(df2 %>% melt(id.vars="np"), aes(x=np-1.55, y=1e2*value,
        color=factor(variable))) + geom_point(size=4) + geom_line(size
        =1)
22
23  data.frame() -> df3
24
25  for(i in seq(0,10,1)){
```

```
26    slanted_grating_simulation_v21(I0=1, slant.angle = i) %>%
27      subset(time==max(time)) %>%
28      distinct(slant.angle,Actual.Shrinkage,Apparent.Shrinkage) %>%
29      rbind(df3) -> df3
30  }
31
32  ggplot(df3 %>% melt(id.vars="slant.angle"), aes(x=slant.angle, y=1
      e2*value, color=factor(variable))) + geom_point(size=4) +
      geom_line(size=1)
```

Fig. 3.19

```
1   data.frame() -> df1
2
3   for(i in seq(10e-4,100e-4,10e-4)){
4     slanted_grating_simulation_v21(T0=i, I0=1, slant.angle = 10) %>%
5       subset(time==max(time)) %>%
6       distinct(T0,Actual.Shrinkage,Apparent.Shrinkage) %>%
7       rbind(df1) -> df1
8   }
9
10  ggplot(df1 %>% melt(id.vars="T0"), aes(x=1e4*T0, y=1e2*value, color
      =factor(variable))) + geom_point(size=4) + geom_line(size=1)
11
12  data.frame() -> df2
13
14  for(i in seq(0,140,20)){
15    slanted_grating_simulation_v21(I0=1, slant.angle = 10, zeta=i)
      %>%
16      subset(time==max(time)) %>%
17      distinct(zeta,Actual.Shrinkage,Apparent.Shrinkage) %>%
18      rbind(df2) -> df2
19  }
```

Fig. 3.20

```
1   data.frame() -> df1
2
3   for(i in c(0,5e-2)){
4     for(j in c(0,139))
5     slanted_grating_simulation_v21(wt.pc=i, zeta=j, slant.angle = 10)
      %>%
6           subset(Y %in% c(0,.5,1)) %>%
7       distinct(time,Delta.n,Y,zeta,wt.pc) %>%
8       rbind(df1) -> df1
9   }
10
```

```
11 ggplot(df1 %>% subset(zeta==0), aes(x=time, y=1e2*Delta.n, color=
      factor(Y), shape=factor(wt.pc))) + geom_point(size=4) +
      geom_line(size=1)

12

13 ggplot(df1 %>% subset(zeta==139), aes(x=time, y=1e2*Delta.n, color=
      factor(Y), shape=factor(wt.pc))) + geom_point(size=4) +
      geom_line(size=1)
```

Fig. 3.21

```
1 data.frame() -> df1

2

3 for(i in c(0,5e-2)){
4   for(j in seq(10e-4,90e-4,20e-4))
5   slanted_grating_simulation_v21(wt.pc=i, T0=j, slant.angle = 10)
      %>%
6     subset(Y %in% c(0,1)) %>%
7     distinct(time,Delta.n,Y,T0,wt.pc) %>%
8     rbind(df1) -> df1
9 }

10

11 ggplot(df1 %>% subset(wt.pc==0), aes(x=time, y=1e2*Delta.n, color=
      factor(T0), shape=factor(Y))) + geom_point(size=4) + geom_line(
      size=1)

12

13 ggplot(df1 %>% subset(wt.pc==5e-2), aes(x=time, y=1e2*Delta.n,
      color=factor(T0), shape=factor(Y))) + geom_point(size=4) +
      geom_line(size=1)
```

Fig. 3.22

```
1 data.frame() -> df1

2

3 for(i in c(0,5e-2)){
4   for(j in c(-10,0,10))
5   slanted_grating_simulation_v21(wt.pc=i, slant.angle = j) %>%
6     subset(Y %in% c(0,1)) %>%
7     distinct(time,Delta.n,Y,slant.angle,wt.pc) %>%
8     rbind(df1) -> df1
9 }

10

11 ggplot(df1 %>% subset(wt.pc==0), aes(x=time, y=1e2*Delta.n, color=
      factor(slant.angle), shape=factor(Y))) + geom_point(size=4) +
      geom_line(size=1)

12

13 ggplot(df1 %>% subset(wt.pc==5e-2), aes(x=time, y=1e2*Delta.n,
      color=factor(slant.angle), shape=factor(Y))) + geom_point(size
      =4) + geom_line(size=1)
```

Fig. 3.23

```r
data.frame() -> df1

for(i in c(0,5e-2)){
  for(j in seq(10e-4,90e-4,20e-4))
  slanted_grating_simulation_v21(wt.pc=i, T0 = j) %>%
    subset(Y %in% c(0,1)) %>%
    distinct(time,delta,Y,T0,wt.pc) %>%
    rbind(df1) -> df1
}

ggplot(df1 %>% subset(wt.pc==0), aes(x=time, y=1e10*delta, color=
    factor(T0), shape=factor(Y))) + geom_point(size=4) + geom_line(
    size=1)

ggplot(df1 %>% subset(wt.pc==5e-2), aes(x=time, y=1e8*delta, color=
    factor(T0), shape=factor(Y))) + geom_point(size=4) + geom_line(
    size=1)
```

Fig. 3.24

```r
data.frame() -> df1

for(i in c(0,5e-2)){
  for(j in c(-10,0,10))
  slanted_grating_simulation_v21(wt.pc=i, slant.angle = j) %>%
    subset(Y %in% c(0,1)) %>%
    distinct(time,delta,Y,slant.angle,wt.pc) %>%
    rbind(df1) -> df1
}

ggplot(df1 %>% subset(wt.pc==0), aes(x=time, y=1e10*delta, color=
    factor(slant.angle), shape=factor(Y))) + geom_point(size=4) +
    geom_line(size=1)

ggplot(df1 %>% subset(wt.pc==5e-2), aes(x=time, y=1e8*delta, color=
    factor(slant.angle), shape=factor(Y))) + geom_point(size=4) +
    geom_line(size=1)
```

Chapter 4

Fig 4.1

```r
sensor_1D_model_v1(sim_holo_grat = coupled_cross_diffusion_v7(),
    exposure_time = 180, output_time_step = 60, a0=0) %>% distinct(x
    ,ze,zs,z0,time) -> df1
```

```r
3 ggplot(df1, aes(x=x, y=ze/z0, color=factor(time))) + geom_lisr(size
    =2)

4

5 ggplot(df1, aes(x=x, y=zs/z0, color=factor(time))) + geom_lisr(size
    =2)
```

Fig. 4.2

```r
1 data.frame(lpmm=as.numeric(), T0=as.numeric(), Klein.Cook=
    as.numeric(), Moharam.Young=as.numeric()) -> df20

2

3 for(i in seq(1e2,1e3,50)){
4   for(j in seq(10e-4,100e-4,10e-4)){
5     slanted_grating_simulation_v21(lpmm=i, T0=j, slant.angle=10,
    total.time=80, F0=0.1*1**0.3) %>% subset(time==max(time) & Y==1)
     %>% distinct(lpmm,T0,Klein.Cook,Moharam.Young) %>% rbind(df20)
    -> df20
6   }
7 }; rm(i,j)

8

9 df20 %>% pull(lpmm) %>% unique %>% sort -> lpmm.vals
10 df20 %>% pull(T0) %>% unique %>% sort -> thickness.vals

11

12 heatmap_matrix=data.frame(lpmm=lpmm.vals)

13

14 for(j in thickness.vals){

15

16   arrange(subset(df20, T0==j), lpmm) %>% select(lpmm,Klein.Cook) ->
     temp

17

18   temp[, as.character(j)] = temp$Klein.Cook

19

20   temp %>% select(-Klein.Cook,-lpmm) -> temp

21

22   heatmap_matrix=cbind(heatmap_matrix,temp)

23

24   rm(temp)

25

26 }; rm(j)

27

28 row.names(heatmap_matrix)=format(lpmm.vals)
29 heatmap_matrix %>% select(-lpmm) -> heatmap_matrix
30 as.matrix(heatmap_matrix) -> heatmap_matrix

31

32 filled.contour(
33   y=1e4*as.numeric(thickness.vals),
34   x=as.numeric(lpmm.vals),
35   z=heatmap_matrix,
36   nlevels=20,
37   main=bquote("Klein-Cook"),
```

```
38    color.palette = function(n) hcl.colors(n, "RdBu"),
39    plot.axes = {
40      axis(1, seq(1e2,1e3,50))
41      axis(2, seq(10e-4,100e-4,10e-4))
42      contour(
43        y=seq(10e-4,100e-4,10e-4),
44        x=seq(1e2,1e3,50),
45        z=heatmap_matrix,
46        nlevels=20,
47        add = TRUE,
48        lwd = 2
49      )
50    }
51 )
```

Fig. 4.3

```
1  heatmap_matrix=data.frame(lpmm=lpmm.vals)
2
3  for(j in thickness.vals){
4
5    arrange(subset(df20, T0==j), lpmm) %>% select(lpmm,Moharam.Young)
       -> temp
6
7    temp[, as.character(j)] = temp$Moharam.Young
8
9    temp %>% select(-Moharam.Young,-lpmm) -> temp
10
11   heatmap_matrix=cbind(heatmap_matrix,temp)
12
13   rm(temp)
14
15 }; rm(j)
16
17 row.names(heatmap_matrix)=format(lpmm.vals)
18 heatmap_matrix %>% select(-lpmm) -> heatmap_matrix
19 as.matrix(heatmap_matrix) -> heatmap_matrix
20
21 filled.contour(
22   y=1e4*as.numeric(thickness.vals),
23   x=as.numeric(lpmm.vals),
24   z=heatmap_matrix,
25   nlevels=20,
26   main=bquote("Moharam-Young"),
27   color.palette = function(n) hcl.colors(n, "RdBu"),
28   plot.axes = {
29     axis(1, seq(1e2,1e3,50))
30     axis(2, seq(10e-4,100e-4,10e-4))
31     contour(
32       y=seq(10e-4,100e-4,10e-4),
```

```
33        x=seq(1e2,1e3,50),
34        z=heatmap_matrix,
35        nlevels=20,
36        add = TRUE,
37        lwd = 2
38      )
39    }
40 )
```

Fig. 4.5

```
1  data.frame() -> df1
2
3  for(i in seq(1e2,1e3,1e2)){
4    coupled_cross_diffusion_v7(lpmm=i) %>%
5      subset(time==max(time)) %>%
6      distinct(lpmm,nu) %>%
7      rbind(df1) -> df1
8  }
9
10 ggplot(df1, aes(x=lpmm, y=nu)) + geom_point(size=4) + geom_line(
     size=1)
11
12 data.frame() -> df2
13
14 for(i in seq(40e-4,100e-4,10e-4)){
15   coupled_cross_diffusion_v7(lpmm=475, T0=i) %>%
16     subset(time==max(time)) %>%
17     distinct(T0,nu) %>%
18     rbind(df2) -> df2
19 }
20
21 ggplot(df2, aes(x=1e4*T0, y=nu)) + geom_point(size=4) + geom_line(
     size=1)
22
23 data.frame() -> df3
24
25 for(i in seq(0,7e-2,5e-3)){
26   coupled_cross_diffusion_v7(lpmm=475, wt.pc=i) %>%
27     subset(time==max(time)) %>%
28     distinct(wt.pc,nu) %>%
29     rbind(df3) -> df3
30 }
31
32 ggplot(df3, aes(x=1e2*wt.pc, y=nu)) + geom_point(size=4) +
     geom_line(size=1)
33
34 data.frame() -> df4
35
36 for(i in seq(2,20,2)){
```

```
37  coupled_cross_diffusion_v7(lpmm=475, I0=2) %>%
38    subset(time==max(time)) %>%
39    distinct(I0,end_exp,nu) %>%
40    rbind(df3) -> df3
41 }
42
43 ggplot(df3, aes(x=I0*end_exp/1e3, y=nu)) + geom_point(size=4) +
      geom_line(size=1)
```

Fig. 4.6

```
1  coupled_cross_diffusion_v7(lpmm=475, I0=1) -> grating_01
2
3  data.frame() -> df1
4
5  for(i in seq(0,.4,.1)){
6    sensor_1D_model_v1(sim_holo_grat = grating_01, a0=0, s0=i, tau_c_
     s = 60, tau_e_s = 60, nzs=1.46) %>%
7      distinct(time,eta,s0) %>%
8      rbind(df1) -> df1
9  }
10
11 ggplot(df1, aes(x=time, y=1e2*eta, color=factor(s0))) + geom_point(
      size=4) + geom_line(size=1)
12
13 data.frame() -> df2
14
15 for(i in c(1.46,1.5,1.54,1.58)){
16   sensor_1D_model_v1(sim_holo_grat = grating_01, a0=0, s0=.1, tau_c
     _s = 60, tau_e_s = 60, nzs=i) %>%
17      distinct(time,eta,nzs,nze) %>%
18      rbind(df2) -> df2
19 }
20
21 ggplot(df2, aes(x=time, y=1e2*eta, color=factor(nzs-nze))) +
      geom_point(size=4) + geom_line(size=1)
22
23 data.frame() -> df3
24
25 for(i in c(10,30,60,180)){
26   sensor_1D_model_v1(sim_holo_grat = grating_01, a0=0, s0=.1, tau_c
     _s = i, tau_e_s = 60, nzs=1.46) %>%
27      distinct(time,eta,tau_c_s) %>%
28      rbind(df3) -> df3
29 }
30
31 ggplot(df3, aes(x=time, y=1e2*eta, color=factor(tau_c_s))) +
      geom_point(size=4) + geom_line(size=1)
32
33 data.frame() -> df4
```

```r
34
35  for(i in c(1,10,30,60,180)){
36    sensor_1D_model_v1(sim_holo_grat = grating_01, a0=0, s0=.1, tau_c
      _s = 60, tau_e_s = i, nzs=1.46) %>%
37      distinct(time,eta,tau_e_s) %>%
38      rbind(df4) -> df4
39  }
40
41  ggplot(df4, aes(x=time, y=1e2*eta, color=factor(tau_e_s))) +
      geom_point(size=4) + geom_line(size=1)
```

Fig. 4.7

```r
1   coupled_cross_diffusion_v7(lpmm=475, I0=1) -> grating_01
2
3   data.frame() -> df1
4
5   for(i in seq(0,.4,.1)){
6     for(j in c(.1,1)){
7       sensor_1D_model_v1(sim_holo_grat = grating_01, a0=i*0.1, s0
      =0.1, tau_c_s = 60, tau_c_a=j*60, tau_e_s = 1e4, tau_e_s=1e4,
      nzs=1.46, nza=1.6, na=2.1, Da=1e-7) %>%
8       distinct(time,eta,s0) %>%
9       rbind(df1) -> df1
10    }
11  }
12
13  ggplot(df1 %>% subset(tau_c_a == tau_c_s), aes(x=time, y=1e2*eta,
      color=factor(a0/s0))) + geom_point(size=4) + geom_line(size=1)
14
15  ggplot(df1 %>% subset(tau_c_a < tau_c_s), aes(x=time, y=1e2*eta,
      color=factor(a0/s0))) + geom_point(size=4) + geom_line(size=1)
```

Fig. 4.8

```r
1   slanted_grating_simulation_v21(lpmm=475, I0=1) -> grating_01
2
3   data.frame() -> df1
4
5   grating_01 %>% subset(time==max(time)) %>% pull(T.t) %>% unique()
      -> T1
6   tys=T1*T1/2.3e-5
7
8   for(i in c(1e-4,0.5,1,10)){
9     for(j in c(0.5,1,2,10)){
10      tcs=j*tys
11      tes=i*tcs
12      sensor_2D_model_v2(sim_holo_grat = grating_01, a0=0, s0=0.2,
      tau_c_s = tcs, tau_e_s = tes) %>%
```

```
13        subset(Y==0) %>%
14        distinct(time,Delta_n,tau_y_s,tau_c_s,tau_e_s) %>%
15        rbind(df1) -> df1
16   }
17 }
18
19 ggplot(df1 %>% subset(round(tau_e_s/tau_c_s,1) < 0.1), aes(x=time,
      y=1e2*Delta_n, color=factor(tau_c_s/tau_y_s))) + geom_point(size
      =4) + geom_line(size=1)
20
21 ggplot(df1 %>% subset(round(tau_e_s/tau_c_s,1) == 0.5), aes(x=time,
       y=1e2*Delta_n, color=factor(tau_c_s/tau_y_s))) + geom_point(
      size=4) + geom_line(size=1)
22
23 ggplot(df1 %>% subset(round(tau_e_s/tau_c_s,1) == 1), aes(x=time, y
      =1e2*Delta_n, color=factor(tau_c_s/tau_y_s))) + geom_point(size
      =4) + geom_line(size=1)
24
25 ggplot(df1 %>% subset(round(tau_e_s/tau_c_s,1) == 10), aes(x=time,
      y=1e2*Delta_n, color=factor(tau_c_s/tau_y_s))) + geom_point(size
      =4) + geom_line(size=1)
```

Fig. 4.9

```
1 data.frame(Delta.n=as.numeric(), tau_c_s=as.numeric(), tau_e_s=
      as.numeric()) -> df3
2 pre_exposure_Delta_n=unique(subset(grating_01, time==max(time) & Y
      ==1)$Delta.n)
3
4 for(i in seq(1,9,0.5)){
5   for(j in seq(0,9,0.5)){
6     sensor_2D_model_v2(sim_holo_grat=grating_01,nzs=1.5,tau_c_s=j,
      tau_e_s=i, a0=0, s0=0.3, Ds=2.3e-5, exposure_time=60, output_
      time_step=1) %>% subset(Y==1 & time==max(time)) %>% distinct(
      time,Delta.n,tau_c_s,tau_e_s) %>% rbind(df3) -> df3
7 }; rm(i,j)
8
9 df3 %>% mutate(Change.in.Delta.n=Delta.n-pre_exposure_Delta_n) ->
      df3
10
11 ggplot(data=df3, mapping=aes(x=Tau.c.s, y=Tau.e.s)) + geom_tile(
      mapping=aes(fill=Change.in.Delta.n), color="White")
```

Fig. 4.10

```
1 data.frame(time=as.numeric(), Delta.n=as.numeric(), eta=as.numeric
      (), tau_e_s=as.numeric(), tau_e_a=as.numeric(), tau_c_s=
      as.numeric(), tau_c_a=as.numeric()) -> df4
2
```

```
3 for(i in c(10,1)){
4   for(j in c(0,1,2,5)){
5     sensor_2D_model_v2(sim_holo_grat=grating_01, nzs=1.6, tau_c_s
      =10, tau_e_s=0.1, tau_c_a=i, tau_e_a=0.1+j, a0=0.1, s0=0.1, Ds
      =2.3e-5, Da=1e-5, exposure_time=60,output_time_step=1) %>%
      subset(Y==1) %>% distinct(time, Delta.n, eta, tau_c_s, tau_e_s,
      tau_c_a, tau_e_a) %>% rbind(df4) -> df4
6   }
7 }; rm(i,j)
8
9 ggplot(df4 %>% subset(tau_c_s==tau_c_a), aes(x=time, y=1e2*Delta.n,
      shape=factor(tau_e_a-tau_e_s))) + geom_line(size=1) +
      geom_point(size=4)
10
11 ggplot(df4 %>% subset(tau_c_s > tau_c_a), aes(x=time, y=1e2*Delta.n
      , shape=factor(tau_e_a-tau_e_s))) + geom_line(size=1) +
      geom_point(size=4)
```

Fig. 4.11

```
1 data.frame(time=as.numeric(), Delta.n=as.numeric(), eta=as.numeric
      (), a0=as.numeric()) -> df6
2
3 for(i in c(0,1/80,1/40,1/20,3/40,1/10)){
4   sensor_2D_model_v2(sim_holo_grat=grating_01, nzs=1.5, nza=1.6,
      tau_c_s=10, tau_e_s=0.1, tau_c_a=1, tau_e_a=5, a0=i, s0=0.1, Ds
      =2.3e-5, Da=1e-5, rhoa=9, rhos=1, exposure_time=60, output_time_
      step=1) %>% subset(Y==1) %>% distinct(time,Delta.n,eta,a0) %>%
      rbind(df6) -> df6
5 }; rm(i)
6
7 ggplot(df6, aes(x=time, y=1e2*Delta.n, shape=factor(a0))) +
      geom_line(size=1) + geom_point(size=4)
8
9 data.frame(time=as.numeric(), Delta.n=as.numeric(), eta=as.numeric
      (), nza=as.numeric(), nze=as.numeric()) -> df7
10
11 for(i in c(1.6,1.7,1.8,1.9,2)){
12  sensor_2D_model_v2(sim_holo_grat=grating_01, nzs=1.5, nza=i, tau_
      c_s=10, tau_e_s=0.1, tau_c_a=1, tau_e_a=5, a0=1.25e-2, s0=0.1,
      Ds=2.3e-5, Da=1e-5, rhoa=9, rhos=1, exposure_time=60, output_
      time_step=1) %>% subset(Y==1) %>% distinct(time,Delta.n,eta,nza,
      nze) %>% rbind(df7) -> df7
13 }; rm(i)
14
15 ggplot(df7, aes(x=time, y=1e2*Delta.n, shape=factor(nza-nze))) +
      geom_line(size=1) + geom_point(size=4)
```

Fig. 4.13

```r
1 slanted_grating_simulation_v15(lpmm=3e2, F0=0.1*1**0.30, Gamma=5e
    -2, T0=30e-4, zeta=139) -> Hybrid5_Thickness30_300lpmm_Angle0_
    Gamma0.05
2
3 data.frame(time=as.numeric(), Thickness=as.numeric(), theta_B_t=
    as.numeric(), s0=as.numeric()) -> df8
4
5 for(i in c(0,0.1,0.2,0.3,0.4)){
6   sensor_2D_model_v2(sim_holo_grat=Hybrid5_Thickness30_300lpmm_
    Angle0_Gamma0.05, a0=0, s0=i, exposure_time=3, output_time_step
    =0.1) %>% subset(Y==1) %>% distinct(time,Thickness,theta_B_t,s0)
     %>% rbind(df8) -> df8
7 }; rm(i)
8
9 ggplot(df8, aes(x=time, y=1e4*Thickness, shape=factor(s0))) +
    geom_line(size=1) + geom_point(size=4)
10
11 ggplot(df8, aes(x=time, y=theta_B_t, shape=factor(s0))) + geom_line
    (size=1) + geom_point(size=4)
```

Fig. 4.14

```r
1 slanted_grating_simulation_v15(start_exp=0, end_exp=1e2, ttotal
    =100, lpmm=1e3, F0=0.1*5**0.3, slant.angle=10, xi=0.3, wt.pc=5e
    -2, epsilon.mz=11, epsilon.pz=11, epsilon.zq=11, T0=50e-4, zeta
    =139) -> grating_01
2
3 data.frame(time=as.numeric(), Thickness=as.numeric(), theta_B_t=
    as.numeric(), s0=as.numeric()) -> df1
4
5 for(i in seq(0,1,0.2)){
6   sensor_2D_model_v2(sim_holo_grat=grating_01,tau_c_s=1.09, tau_e_s
    =1.09, a0=0, s0=i, Ds=2.3e-5, exposure_time=4, output_time_step
    =0.1) %>% subset(Y==1) %>% distinct(time,Thickness,theta_B_t,s0)
     %>% rbind(df1) -> df1
7 }; rm(i)
8
9 ggplot(df1, aes(x=a0/s0, y=1e4*Thickness, shape=factor(s0))) +
    geom_line(size=1) + geom_point(size=4)
10
11 ggplot(df1, aes(x=a0/s0, y=theta_B_t, shape=factor(s0))) +
    geom_line(size=1) + geom_point(size=4)
```

Fig. 4.16

```r
1 planar_grating=slanted_grating_simulation_v21(lpmm=3e2, T0=30e-4)
2
```

```
3 df = data.frame()
4
5 for(i in seq(0,0.4,0.1)){
6
7   sensor_2D_model_v2(sim_holo_grat = planar_grating, a0=i*0.4, s0
      =0.4, nzs=1.46, Ds=5e-8, tau_c_s = 10, tau_e_s = 20, Da=1e-8,
      nza=1.5, tau_c_a=1, tau_e_a=1e5, exposure_time = 180, output_
      time_step = 1) %>% subset(Y==0) %>% distinct(time,eta,a0,s0,nze,
      nza,tau_y_a,tau_y_s,tau_c_a,tau_c_s,tau_e_a,tau_e_s) %>% rbind(
      df) -> df
8
9   sensor_2D_model_v2(sim_holo_grat = planar_grating, a0=i*0.4, s0
      =0.4, nzs=1.46, Ds=5e-8, tau_c_s = 10, tau_e_s = 20, Da=1e-8,
      nza=1.8, tau_c_a=10, tau_e_a=20, exposure_time = 180, output_
      time_step = 1) %>% subset(Y==0) %>% distinct(time,eta,a0,s0,nze,
      nza,tau_y_a,tau_y_s,tau_c_a,tau_c_s,tau_e_a,tau_e_s) %>% rbind(
      df) -> df
10
11  sensor_2D_model_v2(sim_holo_grat = planar_grating, a0=i*0.4, s0
      =0.4, nzs=1.46, Ds=5e-8, tau_c_s = 10, tau_e_s = 20, Da=1e-8,
      nza=1.8, tau_c_a=1, tau_e_a=1e5, exposure_time = 180, output_
      time_step = 1) %>% subset(Y==0) %>% distinct(time,eta,a0,s0,nze,
      nza,tau_y_a,tau_y_s,tau_c_a,tau_c_s,tau_e_a,tau_e_s) %>% rbind(
      df) -> df
12
13  sensor_2D_model_v2(sim_holo_grat = planar_grating, a0=i*0.4, s0
      =0.4, nzs=1.46, Ds=5e-8, tau_c_s = 10, tau_e_s = 20, Da=1e-8,
      nza=1.8, tau_c_a=10, tau_e_a=1e5, exposure_time = 180, output_
      time_step = 1) %>% subset(Y==0) %>% distinct(time,eta,a0,s0,nze,
      nza,tau_y_a,tau_y_s,tau_c_a,tau_c_s,tau_e_a,tau_e_s) %>% rbind(
      df) -> df
14
15 }
16
17 ggplot(data=subset(df, nza==1.5), mapping=aes(x=time, y=1e2*eta,
      color=factor(a0/s0))) + geom_line(size=2)
18
19 ggplot(data=subset(df, nza==1.8 & tau_c_a==tau_c_s & tau_e_a==tau_e
      _s), mapping=aes(x=time, y=1e2*eta, color=factor(a0/s0))) +
      geom_line(size=2)
20
21 ggplot(data=subset(df, nza==1.8 & tau_c_a==tau_c_s & tau_e_a==1e5),
       mapping=aes(x=time, y=1e2*eta, color=factor(a0/s0))) +
      geom_line(size=2)
22
23 ggplot(data=subset(df, nza==1.8 & tau_c_a < tau_c_s & tau_e_a==1e5)
      , mapping=aes(x=time, y=1e2*eta, color=factor(a0/s0))) +
      geom_line(size=2)
```