Chapter 9
Propagation through Atmospheric Turbulence

Up to this point, the propagation algorithms have been designed to simulate propagation through vacuum and through simple optical systems that can be described by ray matrices. There are several other more complicated and useful applications of the split-step beam propagation method. These include sources with partial temporal and spatial coherence, coherent propagation through deterministic structures like fibers and integrated optical devices, and propagation through random media like atmospheric turbulence. This chapter focuses on coherent propagation through turbulence, and the method is shown to be very closely related to propagation through vacuum.

Earth's atmosphere is a medium whose refractive index is nearly unity. This allows us to make only a slight modification to our vacuum-propagation techniques from Ch. 8 to simulate propagation through the atmosphere. Unfortunately, the atmosphere's refractive index randomly evolves over space and time. This effect causes light to be randomly distorted as it propagates. As a result, optical systems that rely on light propagating through the atmosphere must overcome a great challenge. For example, astronomers have observed for centuries that atmospheric turbulence limits the resolution of their telescopes. This is why observatories are built on mountain tops; the location minimizes the turbulent path distance through which the light must propagate.

To simulate atmospheric propagation, we first develop the simulation algorithm, and then we discuss atmospheric turbulence and how to model its refractive properties. Finally, we discuss setting up an atmospheric simulation, proper sampling with due consideration to the effects of the atmosphere, and verifying that the output is consistent with analytic theory.

9.1 Split-Step Beam Propagation Method

Simulating propagation through non-vacuum media is accomplished through the split-step beam propagation method.[40,57–59] This method is useful for simulating propagation through many types of materials: inhomogeneous, anisotropic, and

nonlinear. In this chapter, the discussion is restricted to the atmosphere, which is a linear, isotropic material with inhomogeneous refractive index n, i.e., $n = n(x, y, z)$. When $\delta n = n - 1$ is small, it can be shown that the field in the $i + 1^{\text{st}}$ plane is[59]

$$U(\mathbf{r}_{i+1}) \simeq \mathcal{R}\left[\frac{\Delta z_i}{2}, \mathbf{r}_i, \widetilde{\mathbf{r}}_{i+1}\right] \mathcal{T}[z_i, z_{i+1}] \mathcal{R}\left[\frac{\Delta z_i}{2}, \mathbf{r}_i, \widetilde{\mathbf{r}}_{i+1}\right] \{U(\mathbf{r}_i)\}, \quad (9.1)$$

where $\mathcal{T}[z_i, z_{i+1}]$ is an operator representing the accumulation of phase and $\widetilde{\mathbf{r}}_{i+1}$ is a coordinate in a plane half-way between the i^{th} and $i + 1^{\text{st}}$ planes. It is given by

$$\mathcal{T}[z_i, z_{i+1}] = \exp[-i\phi(\mathbf{r}_{i+1})], \quad (9.2)$$

where the accumulated phase is $\phi(\mathbf{r}_i) = k \int_{z_i}^{z_{i+1}} \delta n(\mathbf{r}_i)\, dz$. Equation (9.1) indicates that we can separate propagation through a medium into two effects: diffraction and refraction. Free-space diffraction is represented by the operator $\mathcal{R}$, while refraction is represented by the operator $\mathcal{T}$. This method is commonly used to simulate propagation though atmospheric turbulence. In fact, it is used to emulate propagation through turbulence in optics laboratories, too.[60,61] The method is to alternate steps of partial vacuum propagation with interaction between the light and the material.[32,43,44]

Writing this algorithm concretely, there is a slight modification to the vacuum propagation algorithm from Eq. (8.18), given by

$$\begin{aligned} U(\mathbf{r}_n) = {} & \mathcal{Q}\left[\frac{m_{n-1} - 1}{m_{n-1}\Delta z_{n-1}}, \mathbf{r}_n\right] \\ & \times \prod_{i=1}^{n-1} \left\{ \mathcal{T}[z_i, z_{i+1}] \mathcal{F}^{-1}\left[\mathbf{f}_i, \frac{\mathbf{r}_{i+1}}{m_i}\right] \mathcal{Q}_2\left[-\frac{\Delta z_i}{m_i}, \mathbf{f}_i\right] \mathcal{F}[\mathbf{r}_i, \mathbf{f}_i] \frac{1}{m_i} \right\} \\ & \times \left\{ \mathcal{Q}\left[\frac{1 - m_1}{\Delta z_1}, \mathbf{r}_1\right] \mathcal{T}[z_1, z_2] U(\mathbf{r}_1) \right\}. \end{aligned} \quad (9.3)$$

Recall that there are $n-1$ propagations and n planes with interaction in each plane. MATLAB code for this algorithm is given in the `ang_spec_multi_prop` function, provided in Listing 9.1. Note that it can be used for vacuum propagation if $\mathcal{T} = 1$ at every step. Example of usage of the `ang_spec_multi_prop` function is given in Sec. 9.5.4 after a discussion of turbulence and how to generate realizations of $\mathcal{T}$.

9.2 Refractive Properties of Atmospheric Turbulence

In this section, the basic theory of atmospheric turbulence is presented. It begins with the original analysis of turbulent flow by Kolmogorov, which eventually led to statistical models of the refractive-index variation.[62] Then, perturbation theory is used with the model to solve Maxwell's equations to obtain useful statistical

Listing 9.1 Code for evaluating the Fresnel diffraction integral in MATLAB through a weakly refractive medium using the angular-spectrum method.

```
function [xn yn Uout] = ang_spec_multi_prop ...
    (Uin, wvl, delta1, deltan, z, t)
% function [xn yn Uout] = ang_spec_multi_prop ...
%     (Uin, wvl, delta1, deltan, z, t)

    N = size(Uin, 1);   % number of grid points
    [nx ny] = meshgrid((-N/2 : 1 : N/2 - 1));
    k = 2*pi/wvl;    % optical wavevector
    % super-Gaussian absorbing boundary
    nsq = nx.^2 + ny.^2;
    w = 0.47*N;
    sg = exp(-nsq.^8/w^16); clear('nsq', 'w');

    z = [0 z];  % propagation plane locations
    n = length(z);
    % propagation distances
    Delta_z = z(2:n) - z(1:n-1);
    % grid spacings
    alpha = z / z(n);
    delta = (1-alpha) * delta1 + alpha * deltan;
    m = delta(2:n) ./ delta(1:n-1);
    x1 = nx * delta(1);
    y1 = ny * delta(1);
    r1sq = x1.^2 + y1.^2;
    Q1 = exp(i*k/2*(1-m(1))/Delta_z(1)*r1sq);
    Uin = Uin .* Q1 .* t(:,:,1);
    for idx = 1 : n-1
        % spatial frequencies (of i^th plane)
        deltaf = 1 / (N*delta(idx));
        fX = nx * deltaf;
        fY = ny * deltaf;
        fsq = fX.^2 + fY.^2;
        Z = Delta_z(idx);   % propagation distance
        % quadratic phase factor
        Q2 = exp(-i*pi^2*2*Z/m(idx)/k*fsq);
        % compute the propagated field
        Uin = sg .* t(:,:,idx+1) ...
            .* ift2(Q2 ...
            .* ft2(Uin / m(idx), delta(idx)), deltaf);
    end
    % observation-plane coordinates
    xn = nx * delta(n);
    yn = ny * delta(n);
    rnsq = xn.^2 + yn.^2;
    Q3 = exp(i*k/2*(m(n-1)-1)/(m(n-1)*Z)*rnsq);
    Uout = Q3 .* Uin;
```

properties of the observation-plane optical field. The variances, correlations, and spectral densities of properties like log-amplitude, phase, and irradiance are used for two primary purposes in conjunction with the simulations. The first use is to produce random draws of the interaction factor for the split-step beam propagation method, which is done in Sec. 9.3. Then, after simulating propagation through the turbulent medium, the observation-plane fields are processed to determine their statistical properties and compare them against theory in Sec. 9.5.5. This provides confirmation that the simulation is producing accurate results.

9.2.1 Kolmogorov theory of turbulence

Turbulence in Earth's atmosphere is caused by random variations in temperature and convective air motion, which alter the air's refractive index, both spatially and temporally. As optical waves propagate through the atmosphere, the waves are distorted by these fluctuations in refractive index. This distortion of light has frustrated astronomers for centuries because it degrades their images of celestial objects. To overcome this distortion, they needed an accurate physical model of turbulence and its effects on optical-wave propagation. Since turbulence affects all optical systems that rely on propagating light through long atmospheric paths, like laser communication systems and laser weapons, optical physicists and communications engineers have begun to address this problem more recently.

Over the last hundred years, modeling the effects of turbulence on optical propagation has received much attention. Much has been written on various theories and experimental verification thereof. The focus on statistical modeling has produced several useful theories. In these theories, it is necessary to resort to statistical analyses, because it is impossible to exactly describe the refractive index for all positions in space and all time. There are too many random behaviors and variables to account for in a closed-form solution. The most widely accepted theory of turbulent flow, due to its consistent agreement with observation, was first put forward by A. N. Kolmogorov.[62] Later, Obukhov[63] and independently Corrsin[64] adapted Kolmogorov's model to temperature fluctuations. Then, the theory of turbulent temperature fluctuations could be directly related to refractive-index fluctuations. This model is the basis for all contemporary theories of turbulence.[65]

Differential heating and cooling of Earth by sunlight and the diurnal cycle cause large-scale variations in the temperature of air. This process consequently creates wind. As air moves, it transitions from laminar flow to turbulent flow. In laminar flow, the velocity characteristics are uniform or at least change in a regular fashion. In turbulent flow, air of different temperatures mixes, so the velocity field is no longer uniform, and it acquires randomly distributed pockets of air, called turbulent eddies. These eddies have varying characteristic sizes and temperatures. Since the density of air, and thus its refractive index, depends on temperature, the atmosphere has a random refractive-index profile.

Turbulent flow is a nonlinear process governed by the Navier-Stokes equations.

Because there are difficulties in solving the Navier-Stokes equations for fully developed turbulence, Kolmogorov developed a statistical theory. He suggested that in turbulent flow, the kinetic energy in large eddies is transferred into smaller eddies. The average size of the largest eddies, L_0, is called the outer scale. Near the ground, L_0 is on the order of the height above ground, while high above the ground, it can be just tens to hundreds of meters.[66] The average size of the smallest turbulent eddies, l_0, is called the inner scale. At very small scales, smaller than the inner scale, the energy dissipation caused by friction prevents the turbulence from sustaining itself. The inner scale l_0 can be a few millimeters near the ground to a few centimeters high above the ground.[66] The range of eddy sizes between the inner and outer scales is called the inertial subrange.

In Kolmogorov's analysis, he assumed that eddies within the inertial subrange are statistically homogeneous and isotropic within small regions of space, meaning that properties like velocity and refractive index have stationary increments. This was the reason for using the structure function rather than the more common covariance. It allowed him to use dimensional analysis to determine that the average speed of turbulent eddies v must be related to the scale size of eddies, r, via[62]

$$v \propto r^{1/3}. \tag{9.4}$$

Then, since the structure function of speed is a square of speeds, the structure function $D_v(r)$ must follow the form

$$D_v(r) = C_v^2 r^{2/3}, \tag{9.5}$$

where C_v is the velocity structure parameter. For laminar flow, which occurs at very small scales, the physical dependencies are slightly different, so the velocity structure function follows the form

$$D_v(r) = C_v^2 l_0^{-4/3} r^2. \tag{9.6}$$

For the largest scales of turbulence, the flow is highly anisotropic. If the velocity field was homogeneous and isotropic, the structure function would asymptotically approach twice the velocity variance.

This velocity framework lead to a similar analysis of potential temperature θ (potential temperature is linearly related to ordinary temperature T). The results are $\theta \propto r^{1/3}$ so that the potential temperature structure function $D_\theta(r)$ follows the same dependence as the velocity structure function, yielding[63,64]

$$D_\theta(r) = \begin{cases} C_\theta^2 l_0^{-4/3} r^2, & 0 \le r \ll l_0 \\ C_\theta^2 r^{2/3}, & l_0 \ll r \ll L_0, \end{cases} \tag{9.7}$$

where C_θ^2 is the structure parameter of θ.

A few more considerations produce a model for refractive-index statistics. Now, the refractive index at a point in space $\mathbf{r}$ can be written as

$$n(\mathbf{r}) = \mu_n(\mathbf{r}) + n_1(\mathbf{r}), \tag{9.8}$$

where $\mu_n(\mathbf{r}) \cong 1$ is the slowly varying mean value of the refractive index, and $n_1(\mathbf{r})$ is the deviation of the index from its mean value. Writing the refractive index this way creates a zero-mean random process $n_1(\mathbf{r})$, which is easier to work with for the following statistical analysis. At optical wavelengths, the refractive index of air is given approximately by

$$n(\mathbf{r}) = 1 + 77.6 \times 10^{-6} \left(1 + 7.52 \times 10^{-3} \lambda^{-2}\right) \frac{P(\mathbf{r})}{T(\mathbf{r})} \tag{9.9}$$

$$\cong 1 + 7.99 \times 10^{-5} \frac{P(\mathbf{r})}{T(\mathbf{r})} \qquad \text{for} \qquad \lambda = 0.5\,\mu\text{m}, \tag{9.10}$$

where λ is the optical wavelength in micrometers, P is the pressure in millibars, and T is the ordinary temperature in Kelvin. The variation in refractive index is given by

$$dn = 7.99 \times 10^{-5} \left(dP - \frac{-dT}{T^2}\right). \tag{9.11}$$

In this model, each eddy is considered to have relatively uniform pressure. Also, the reader should recall that potential temperature θ is linearly related to ordinary temperature T. Therefore, the refractive index variation becomes

$$dn = 7.99 \times 10^{-5} \frac{d\theta}{T^2}. \tag{9.12}$$

Because the variation in refractive index is directly proportional to the variation in potential temperature, the refractive index structure function $D_n(r)$ follows the same power law as $D_\theta(r)$ so that

$$D_n(r) = \begin{cases} C_n^2 \, l_0^{-4/3} \, r^2, & 0 \le r \ll l_0 \\ C_n^2 \, r^{2/3}, & l_0 \ll r \ll L_0, \end{cases} \tag{9.13}$$

where C_n^2 is known as the refractive-index structure parameter, measured in $\text{m}^{-2/3}$. It is related to the temperature structure constant by

$$C_n^2 = \left[77.6 \times 10^{-6} \left(1 + 7.52 \times 10^{-3} \lambda^{-2}\right) \frac{P}{T^2}\right]^2 C_T^2. \tag{9.14}$$

Typical values of C_n^2 are in the range 10^{-17}–10^{-13} $\text{m}^{-2/3}$, with small values at high altitudes and large values near the ground.

It is often necessary to have a spectral description of refractive-index fluctuations. The power spectral density $\Phi_n(\kappa)$ can easily be computed from Eq. (9.13)

and vice versa.[15] For example, the Kolmogorov refractive-index power spectral density is computed by

$$\Phi_n^K(\kappa) = \frac{1}{4\pi^2\kappa^2}\int_0^\infty \frac{\sin(\kappa r)}{\kappa r}\frac{d}{dr}\left[r^2\frac{d}{dr}D_n(r)\right]dr \tag{9.15}$$

$$= 0.033\,C_n^2\kappa^{-11/3} \qquad \text{for} \qquad \frac{1}{L_0} \ll \kappa \ll \frac{1}{l_0}, \tag{9.16}$$

where $\boldsymbol{\kappa} = 2\pi\left(f_x\hat{\mathbf{i}} + f_y\hat{\mathbf{j}}\right)$ is angular spatial frequency in rad/m. The reader should note that Eq. (9.15) is valid only for random fields that are locally homogeneous and isotropic.

There are other models for the refractive power spectral density, like the Tatarskii, von Kármán, modified von Kármán, and Hill spectrum, which are commonly used.[15] These are each more sophisticated and include various inner-scale and outer-scale factors that improve the agreement between theory and experimental measurements. These power spectra are shown in Fig. 9.1. Two of the simplest practical models are the von Kármán PSD, given by

$$\Phi_n^{vK}(\kappa) = \frac{0.033\,C_n^2}{\left(\kappa^2+\kappa_0^2\right)^{11/6}} \qquad \text{for} \qquad 0 \le \kappa \ll 1/l_0, \tag{9.17}$$

and the modified von Kármán PSD

$$\Phi_n^{mvK}(\kappa) = 0.033\,C_n^2\frac{\exp\left(-\kappa^2/\kappa_m^2\right)}{\left(\kappa^2+\kappa_0^2\right)^{11/6}} \qquad \text{for} \qquad 0 \le \kappa < \infty, \tag{9.18}$$

where $\kappa_m = 5.92/l_0$ and $\kappa_0 = 2\pi/L_0$. The values of κ_m and κ_0 are chosen to match the small-scale (high-frequency) and large-scale (low-frequency) behavior predicted by the dimensional analysis. The modified von Kármán is the simplest PSD model that includes effects of both inner and outer scales.[15] Note that when $l_0 = 0$ and $L_0 = \infty$ are used, Eq. (9.18) reduces to Eq. (9.16).

When dealing with electromagnetic propagation through the atmosphere, the refractive index can be considered independent of time over short (100 μs) time scales. Because the speed of light is so fast, the time it takes light to traverse even a very large turbulent eddy is much, much shorter than the time it takes for an eddy's properties to change. Consequently, temporal properties are built into turbulence models through the Taylor frozen-turbulence hypothesis. The hypothesis is that temporal variations in meteorological quantities at a location in space are caused by advection of these quantities by the mean-speed wind flow, not by changes in the quantities themselves.[15] Consequently, turbulent eddies are treated as frozen in space and blown across the optical axis by the mean wind velocity $\mathbf{v}$. Then, with knowledge of the mean wind speed, one converts spatial statistics into temporal

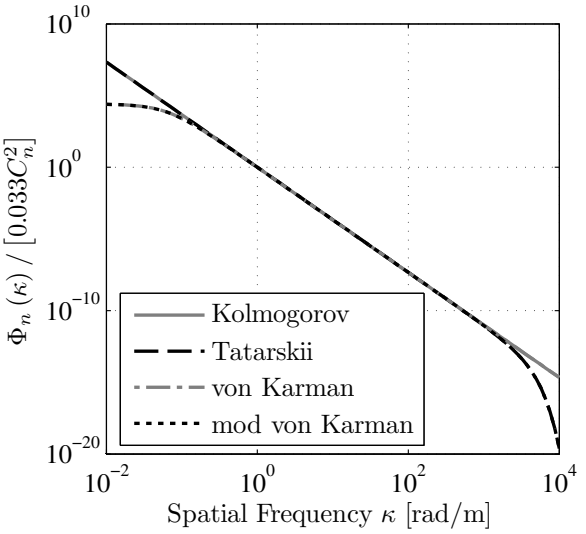

Figure 9.1 Common models for atmospheric power spectra.

statistics. For example, the temporal dependence of optical phase $\phi(x, y)$ is given by

$$\phi(x, y, t) = \phi(x - v_x t, y - v_y t, 0), \tag{9.19}$$

where v_x and v_y are the Cartesian components of the mean wind velocity, and t is time.

9.2.2 Optical propagation through turbulence

As described in Ch. 1, electromagnetic phenomena are governed by Maxwell's equations for both vacuum and atmospheric turbulence. The atmosphere may be considered a source-free, nonmagnetic, and isotropic medium. For optical-wave propagation, we seek solutions of a traveling wave with harmonic time dependence $\exp(-i2\pi\nu t)$, where $\nu = c/\lambda$ is the frequency of the light just like in Sec. 1.2.1. Then, the wave equation for the electric field may be written as[15]

$$\nabla^2 \mathbf{E}(\mathbf{r}) + k^2 n^2(\mathbf{r}) \mathbf{E}(\mathbf{r}) + 2\nabla [\mathbf{E}(\mathbf{r}) \cdot \nabla \ln n(\mathbf{r})] = \mathbf{0}, \tag{9.20}$$

where $\mathbf{E}$ is the electric field vector and k is the vacuum optical wavenumber. The last term in Eq. (9.20) refers to the change in polarization as the wave propagates. It can be neglected for $\lambda < l_0$, and consequently the wave equation simplifies to

$$\left[\nabla^2 + k^2 n^2(\mathbf{r})\right] \mathbf{E}(\mathbf{r}) = \mathbf{0}. \tag{9.21}$$

Like in Sec. 1.2.1, the magnetic induction $\mathbf{B}$ obeys this equation, too, so we can write one equation for any of the six field components:

$$\left[\nabla^2 + k^2 n^2(\mathbf{r})\right] U(\mathbf{r}) = 0. \tag{9.22}$$

This is almost identical to Eq. (1.43), except that the refractive index is explicitly position-dependent here. In solving Eq. (9.22), we recall Eq. (9.8) and assume that $|n_1(\mathbf{r})| \ll 1$. This is the assumption of weak fluctuations, which is quantified later in this chapter. With this approximation, the factor $n^2(\mathbf{r})$ in Eq. (9.22) can be approximated by

$$n^2(\mathbf{r}) \cong 1 + 2n_1(\mathbf{r}). \tag{9.23}$$

Then, the wave equation becomes

$$\left\{\nabla^2 + k^2\left[1 + 2n_1(\mathbf{r})\right]\right\} U(\mathbf{r}) = 0. \tag{9.24}$$

When the medium has a constant index of refraction, Eq. (9.22) is solved by the methods of Fourier optics from Sec. 1.3, which involve the use of Green's functions. However, when the medium is randomly inhomogeneous, as is the case with the atmosphere, perturbative methods are used with Green's functions to obtain approximate solutions. In the Rytov method, the optical field is written as

$$U(\mathbf{r}) = U_0(\mathbf{r}) \exp\left[\psi(\mathbf{r})\right], \tag{9.25}$$

where $U_0(\mathbf{r})$ is the vacuum solution ($n_1 = 0$) of Eq. (9.24), and $\psi(\mathbf{r})$ is the complex phase perturbation. The form

$$\psi(\mathbf{r}) = \psi_1(\mathbf{r}) + \psi_2(\mathbf{r}) + \ldots \tag{9.26}$$

is used to perform successive perturbations. These successive perturbations are used to compute various statistical moments of ψ which, in turn, yield statistical moments of the field. Further, it is useful to isolate amplitude and phase quantities by writing

$$\psi = \chi + i\phi, \tag{9.27}$$

where χ is the log-amplitude perturbation, and ϕ is the phase perturbation. The Rytov method can be used with a given PSD model to analytically compute moments of the field for simple source fields like Gaussian beams, spherical waves, and plane waves. The reader is referred to Clifford,[67] Ishimaru,[65] Andrews and Phillips,[15] and Sasiela[68] for greater detail about the Rytov method.

9.2.3 Optical parameters of the atmosphere

The details of the derivations are omitted here, but useful field moments that can be calculated from Rytov theory include

- the mean value of the optical field

$$\langle U(\mathbf{r})\rangle = U_0(\mathbf{r}) \langle \exp\psi(\mathbf{r})\rangle, \quad \text{and} \tag{9.28}$$

- the mutual coherence function

$$\Gamma\left(\mathbf{r},\mathbf{r}',z\right) = \left\langle U\left(\mathbf{r}\right)U^{*}\left(\mathbf{r}'\right)\right\rangle \tag{9.29}$$

$$= U_0\left(\mathbf{r}\right)U_0^{*}\left(\mathbf{r}'\right)\left\langle \exp\left[\psi\left(\mathbf{r}\right)\psi^{*}\left(\mathbf{r}'\right)\right]\right\rangle . \tag{9.30}$$

From the mutual coherence function, we can compute many useful properties, including

- the modulus of the complex coherence factor (hereafter called the coherence factor)[6]

$$\mu\left(\mathbf{r},\mathbf{r}',z\right) = \frac{\left|\Gamma\left(\mathbf{r},\mathbf{r}',z\right)\right|}{\left[\Gamma\left(\mathbf{r},\mathbf{r},z\right)\Gamma\left(\mathbf{r}',\mathbf{r}',z\right)\right]^{1/2}}, \tag{9.31}$$

- the wave structure function

$$D\left(\mathbf{r},\mathbf{r}',z\right) = -2\ln\mu\left(\mathbf{r},\mathbf{r}',z\right) \tag{9.32}$$

$$= D_{\chi}\left(\mathbf{r},\mathbf{r}',z\right) + D_{\phi}\left(\mathbf{r},\mathbf{r}',z\right), \tag{9.33}$$

  where D_{χ} and D_{ϕ} are the log-amplitude and phase structure functions, respectively,

- the phase power spectral density

$$\Phi_{\phi}\left(\kappa\right) = \frac{1}{4\pi^2\kappa^2}\int_0^{\infty}\frac{\sin\left(\kappa r\right)}{\kappa r}\frac{d}{dr}\left[r^2\frac{d}{dr}D_{\phi}\left(r\right)\right]dr, \quad \text{and} \tag{9.34}$$

- the mean MTF of the turbulent path

$$\mathcal{H}\left(f\right) = \exp\left[-\frac{1}{2}D\left(\lambda f_l f\right)\right], \tag{9.35}$$

  where f_l is the system focal length.

Each of these properties are discussed below. Then later, some of these theoretical properties are used to validate turbulent wave-optics simulations.

The structure parameter C_n^2 is a measure of the local turbulence strength. However, there are other, more useful and measurable quantities that have more intuitive meanings. Additionally, C_n^2 is a function of the propagation distance Δz, so sometimes single numbers are more handy to characterize specific optical effects. Consequently, $C_n^2\left(z\right)$ is commonly used to compute parameters like the atmospheric coherence diameter r_0 and isoplanatic angle θ_0, discussed below. In fact, the coherence diameter and isoplanatic angle are related to integrals of $C_n^2\left(z\right)$.

In the case of an isotropic and homogeneous optical field, the modulus of the coherence factor can be computed as[6][8]

$$\mu\left(\mathbf{r},\mathbf{r}',z\right) = \mu\left(\mathbf{r},\mathbf{r}+\Delta\mathbf{r},z\right) = \mu\left(\Delta\mathbf{r},z\right) = \mu\left(\left|\Delta\mathbf{r}\right|,z\right). \tag{9.36}$$

The exact form of the coherence factor depends on both the type of optical source and the type of refractive-index PSD being used. As a simple example, when the source is a plane wave,

$$\mu\left(\left|\Delta\mathbf{r}\right|,\Delta z\right)=\exp\left\{-4\pi^2k^2\int_0^{\Delta z}\int_0^{\infty}\Phi_n\left(\kappa,z\right)\left[1-J_0\left(\kappa\left|\Delta\mathbf{r}\right|\right)\right]\kappa\,d\kappa dz\right\}, \tag{9.37}$$

and the only dependence on the propagation path is $C_n^2\left(z\right)$ within the refractive-index PSD. When the Kolmogorov spectrum is used, the coherence factor evaluates to

$$\mu^K\left(\left|\Delta\mathbf{r}\right|,\Delta z\right)=\exp\left[-1.46k^2\left|\Delta\mathbf{r}\right|^{5/3}\int_0^{\Delta z}C_n^2\left(z\right)\,dz\right]. \tag{9.38}$$

The spatial coherence radius ρ_0 of an optical wave is defined as the e^{-1} point of $\mu\left(\left|\Delta\mathbf{r}\right|,\Delta z\right)$. Now, recalling Eq. (9.32) allows us to write

$$D\left(\rho_0,z\right)=2\,\text{rad}^2 \tag{9.39}$$

as an equivalent definition of ρ_0. With either definition, the coherence radius for a plane wave in Kolmogorov turbulence is computed as

$$\rho_0=\left[1.46k^2\int_0^{\Delta z}C_n^2\left(z\right)\,dz\right]^{-3/5}. \tag{9.40}$$

The atmospheric coherence diameter r_0 is a more commonly used parameter, and it is given by[15]

$$D\left(r_0,z\right)=6.88\,\text{rad}^2\qquad\text{and}\qquad r_0=2.1\,\rho_0 \tag{9.41}$$

for a plane wave. It also known as the Fried parameter because it was first introduced by D. L. Fried.[69] In fact, it was introduced in a very different way from ρ_0. Fried analyzed the resolution of an imaging telescope as the volume underneath the atmospheric MTF. When written as a function of telescope diameter, the knee in the curve was defined as r_0. For a plane-wave source, the atmospheric coherence diameter $r_{0,pw}$ is mathematically computed as[68]

$$r_{0,pw}=\left[0.423k^2\int_0^{\Delta z}C_n^2\left(z\right)\,dz\right]^{-3/5}, \tag{9.42}$$

where light propagates from the source at $z=0$ to the receiver at $z=\Delta z$. For a point source (spherical wave), the atmospheric coherence diameter $r_{0,sw}$ is computed as[68]

$$r_{0,sw}=\left[0.423k^2\int_0^{\Delta z}C_n^2\left(z\right)\left(\frac{z}{\Delta z}\right)^{5/3}dz\right]^{-3/5}. \tag{9.43}$$

Values of r_0 are typically 5–10 cm for visible wavelengths and vertical viewing.

With these definitions and letting $r = |\Delta\mathbf{r}|$, the wave structure function for a plane-wave source with Kolmogorov turbulence can be written as[15]

$$D^K(r) = 6.88\left(\frac{r}{r_0}\right)^{5/3}. \tag{9.44}$$

Recall that the inner scale and outer scale are assumed to be $l_0 = 0$ and $L_0 = \infty$ in this case. Using the von Kármán PSD, we can account for a finite outer scale, resulting in a more accurate structure function given by

$$D^{vK}(r) = 6.16 r_0^{-5/3}\left[\frac{3}{5}\kappa_0^{-5/3} - \frac{(r/\kappa_0/2)^{5/6}}{\Gamma(11/6)}K_{5/6}(\kappa_0 r)\right]. \tag{9.45}$$

When both the inner and outer scales are important, we can use the modified von Kármán PSD to yield

$$D^{mvK}(r) = 3.08 r_0^{-5/3} \times\left\{\Gamma\left(-\frac{5}{6}\right)\kappa_m^{-5/3}\left[1 -_1 F_1\left(-\frac{5}{6};1;-\frac{\kappa_m^2 r^2}{4}\right)\right] - \frac{9}{5}\kappa_0^{1/3} r^2\right\}, \tag{9.46}$$

where ${}_1F_1(a;c;z)$ is a confluent hypergeometric function of the first kind and the modified von Kármán PSD has been used. Andrews *et al.*[70] presented an algebraic approximation for the hypergeometric function that allows this structure function to be written in the simpler form

$$D^{mvK}(r) \simeq 7.75 r_0^{-5/3} l_0^{-1/3} r^2\left[\frac{1}{\left(1 + 2.03 r^2/l_0^2\right)^{1/6}} - 0.72(\kappa_0 l_0)^{1/3}\right], \tag{9.47}$$

with $< 2\%$ error. The wave structure functions for other sources and more sophisticated PSD models like the Hill model can be found in references like Andrews and Phillips.[15] The plane-wave cases are given here because they are very useful, particularly for verifying the properties of randomly generated phase screens used in wave-optics simulations.

With the various forms of the wave structure function calculated, Eq. (9.34) allows us to compute the phase PSD. Practically speaking though, there is another relationship that makes the phase PSD much easier to calculate. For a plane wave in weak turbulence, the phase PSD is

$$\Phi_\phi(\kappa) = 2\pi^2 k^2 \Delta z \Phi_n(\kappa). \tag{9.48}$$

Then, it is straightforward to show that the phase PSDs for the Kolmogorov, von Kármán, and modified von Kármán refractive-index PSD's are

$$\Phi_\phi^K(\kappa) = 0.49 r_0^{-5/3}\kappa^{-11/3}, \tag{9.49}$$

$$\Phi_\phi^{vK}(\kappa) = \frac{0.49 r_0^{-5/3}}{\left(\kappa^2 + \kappa_0^2\right)^{11/6}}, \tag{9.50}$$

and

$$\Phi_\phi^{mvK}(\kappa) = 0.49 r_0^{-5/3} \frac{\exp\left(-\kappa^2/\kappa_m^2\right)}{\left(\kappa^2 + \kappa_0^2\right)^{11/6}}, \tag{9.51}$$

respectively. Later in the chapter, these PSDs are used to generate random draws of turbulent phase screens. The method makes use of FTs, and this book's FT convention uses ordinary frequency in cycles/m, rather than angular frequency in rad/m. Accordingly, it is useful to write the PSD in terms of f, which yields

$$\Phi_\phi^{K}(f) = 0.023 r_0^{-5/3} f^{-11/3}, \tag{9.52}$$

as one example. The other PSDs follow similarly.

When Fried introduced r_0, he did it as a part of calculating the average MTF of images taken through the atmosphere.[69] His results can be summarized as[6]

$$\mathcal{H}(f) = \exp\left\{-3.44\left(\frac{\lambda f_l f}{r_0}\right)^{5/3}\left[1 - \alpha\left(\frac{\lambda f_l f}{D}\right)^{1/3}\right]\right\} \tag{9.53}$$

$$= \exp\left\{-3.44\left(\frac{f}{2f_0}\frac{D}{r_0}\right)^{5/3}\left[1 - \alpha\left(\frac{f}{2f_0}\right)^{1/3}\right]\right\}, \tag{9.54}$$

where again f_0 is the diffraction-limited cutoff frequency and

$$\alpha = \begin{cases} 0 & \text{for long-exposure imagery,} \\ 1 & \text{for short-exposure imagery without scintillation,} \\ \frac{1}{2} & \text{for short-exposure imagery with scintillation.} \end{cases} \tag{9.55}$$

The key distinction between short exposures and long exposures here lies in the correction of atmospheric tilt. Long-exposure images are assumed to be long enough that the image center wanders randomly many times in the image plane. Conversely, short-exposure images are assumed to be short enough that only one realization of tilt affects the image. When multiple short-exposure images are averaged, the images are first shifted to the center, thereby removing the effects of tilt. The reader should note that the atmosphere has a transfer function given by Eq. (9.54), while the imaging system has its own OTF as discussed in Sec. 5.2.2. The OTF of the composite system is the product of the two OTFs. As an example, a plot of the composite MTFs is shown in Fig. 9.2 for a circular aperture and $D/r_0 = 4$.

As discussed in Sec. 5.2.3, the average MTF can be used to determine an imaging system's Strehl ratio. Fried's work provides a way to include the effects of

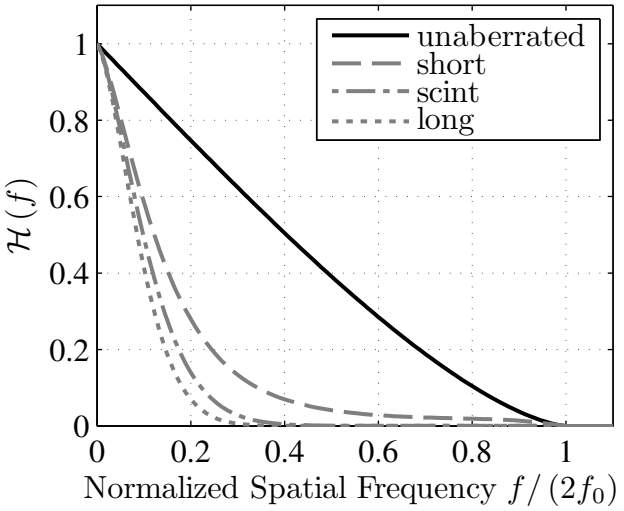

Figure 9.2 Composite MTFs for $D/r_0 = 4$. The solid black line shows the unaberrated case. The gray dashed line shows the short-exposure case with only phase fluctuations. The gray dash-dot line shows the short-exposure case when scintillation is significant. The gray dotted line shows the long-exposure case.

turbulence when calculating Strehl ratio. Making use of Eqs. (5.47) and (9.54), the Strehl ratio for a circular aperture in turbulence is given by

$$\mathcal{S} = \frac{16}{\pi} \int_0^1 f' \left(\cos^{-1} f' - f' \sqrt{1 - f'^2} \right) \times \exp \left\{ -3.44 \left(f' \frac{D}{r_0} \right)^{5/3} \left[1 - \alpha \left(f' \right)^{1/3} \right] \right\} df', \tag{9.56}$$

where $f' = f/(2f_0)$ is normalized spatial frequency. Fried numerically evaluated this integral for each value of α. Later, Andrews and Phillips developed an analytic approximation for the long-exposure case without scintillation ($\alpha = 0$) given by[15]

$$\mathcal{S} \cong \frac{1}{\left[1 + (D/r_0)^{5/3} \right]^{6/5}}. \tag{9.57}$$

Their approximation is quite accurate for all D/r_0. Sasiela evaluated this case of the integral using Mellin transforms, resulting in an expression that can be written either as a Meijer G-function or equivalently as a Fox H-function.[68] Using the first few terms of a series representation leads to the approximate polynomial expression:

$$\mathcal{S} \cong \left(\frac{r_0}{D} \right)^2 - 0.6159 \left(\frac{r_0}{D} \right)^3 + 0.0500 \left(\frac{r_0}{D} \right)^5 + 0.132 \left(\frac{r_0}{D} \right)^7, \tag{9.58}$$

which is extremely accurate for $D/r_0 > 2$.

If an optical system's characteristics (optical transfer function and point-spread function) are not shift-invariant, the system has a property called anisoplanatism. This applies to any optical system, but the system of interest here is the atmosphere. To measure the severity of angular anisoplanatism, we can examine an angular structure function of the phase $D_\phi(\theta)$ defined by

$$D_\phi(\Delta\theta) = \left\langle |\phi(\theta) - \phi(\theta + \Delta\theta)|^2 \right\rangle, \tag{9.59}$$

where θ is an angular coordinate in the object field and $\Delta\theta$ is an angular separation between two points in the object field. The isoplanatic angle θ_0 is defined as the angle for which

$$D_\phi(\theta_0) = 1\,\text{rad}^2. \tag{9.60}$$

By similar mathematics to those that lead to Eq. (9.43), θ_0 is given by

$$\theta_0 = \left[2.91 k^2 \Delta z^{5/3} \int_0^{\Delta z} C_n^2(z) \left(1 - \frac{z}{\Delta z}\right)^{5/3} dz \right]^{-3/5}. \tag{9.61}$$

This may be considered the largest field angle over which the optical path length through the turbulence does not differ significantly from the on-axis optical path length through the turbulence. Values of θ_0 are typically 5–10 μrad for visible wavelengths and vertical viewing.

Log-amplitude (or equivalently, irradiance) statistics are also important to describe the strength of scintillations. The log-amplitude variance, defined as

$$\sigma_\chi^2(\mathbf{r}) = \left\langle \chi^2(\mathbf{r}) \right\rangle - \left\langle \chi(\mathbf{r}) \right\rangle^2, \tag{9.62}$$

is a common measure of scintillation. For plane-wave and diverging spherical-wave (point) sources, the log-amplitude variances $\sigma_{\chi,pw}^2$ and $\sigma_{\chi,sw}^2$ evaluate to[68]

$$\sigma_{\chi,pw}^2 = 0.563 k^{7/6} \Delta z^{5/6} \int_0^{\Delta z} C_n^2(z) \left(1 - \frac{z}{\Delta z}\right)^{5/6} dz \tag{9.63}$$

and

$$\sigma_{\chi,sw}^2 = 0.563 k^{7/6} \int_0^{\Delta z} C_n^2(z)\, z^{5/6} \left(1 - \frac{z}{\Delta z}\right)^{5/6} dz, \tag{9.64}$$

respectively. Weak fluctuations are associated with $\sigma_\chi^2 < 0.25$, and strong fluctuations with $\sigma_\chi^2 \gg 0.25$. Note that the Rytov method presented here is valid only for weak fluctuations.

9.2.4 Layered atmosphere model

Deriving analytic results for atmospheric turbulence effects on optical propagation is possible when we assume a simple statistical model. However, when one wants to consider more complex scenarios like using adaptive-optics systems, usually the statistics of the corrected optical fields cannot be computed in closed form. For mathematical simplification, a common technique is to treat turbulence as a finite number of discrete layers. This approach is common for analytic calculations, computer simulations, and emulating turbulence in the laboratory.[15,60,61] A layered model is useful if its refractive index spectrum and scintillation properties match that of the corresponding extended medium.[23,71]

Each layer is a unit-amplitude thin phase screen which represents a turbulent volume of a much greater thickness. A phase screen is considered thin if its thickness is much less than the propagation distance following the screen.[15] A phase screen is one realization of an atmospheric phase perturbation, and it is used with Eq. (9.2) to compute a realization of the refraction operator $\mathcal{T}\left[z_i, z_{i+1}\right]$. This is how atmospheric phase screens are incorporated into the split-step beam propagation method to simulate atmospheric propagation. A discussion of layered turbulence theory and phase screen generation follows.

9.2.5 Theory

To theoretically represent the atmosphere as phase screens, we simply write the turbulence profile in terms of the effective structure parameter $C_{n_i}^2$, the location along the propagation path z_i, and the thickness Δz_i of the slab of extended turbulence represented by the i^{th} phase screen. The values of $C_{n_i}^2$ are chosen so that several low-order moments of the continuous model match the layered model:[23,71]

$$\int_0^{\Delta z} C_n^2\left(z'\right)\left(z'\right)^m \, dz' = \sum_{i=1}^{n} C_{n_i}^2 z_i^m \Delta z_i, \tag{9.65}$$

where n is the number of phase screens being used, and $0 \leq m \leq 7$. This way, r_0, θ_0, σ_χ^2, etc. of the layered model match the parameters of the bulk turbulence being modeled. The atmospheric parameters for the layered turbulence model are computed using the discrete-sum versions of Eqs. (9.42), (9.43), (9.63), and (9.64) given by

$$r_{0,pw} = \left(0.423k^2 \sum_i C_{n_i}^2 \, \Delta z_i\right)^{-3/5} \tag{9.66}$$

$$r_{0,sw} = \left[0.423k^2 \sum_{i=1}^{n} C_{n_i}^2 \left(\frac{z_i}{\Delta z}\right)^{5/3} \Delta z_i\right]^{-3/5} \tag{9.67}$$

$$\sigma_{\chi,pw}^2 = 0.563k^{7/6}\Delta z^{5/6} \sum_{i=1}^{n} C_{n_i}^2 \left(1 - \frac{z_i}{\Delta z}\right)^{5/6} \Delta z_i \tag{9.68}$$

$$\sigma^2_{\chi,sw} = 0.563k^{7/6}\Delta z^{5/6}\sum_{i=1}^{n} C^2_{n_i}\left(\frac{z_i}{\Delta z}\right)^{5/6}\left(1-\frac{z_i}{\Delta z}\right)^{5/6}\Delta z_i. \tag{9.69}$$

By grouping terms in Eq. (9.66), the i^{th} layer can be given an effective coherence diameter r_{0_i} given by[71]

$$r_{0_i} = \left[0.423\,k^2\,C^2_{n_i}\Delta z_i\right]^{-3/5}. \tag{9.70}$$

Note that this is the plane-wave r_0, so it is valid only when the layer is very thin. The r_0 values for turbulence layers are commonly used for characterizing their strength. With this definition, Eq. (9.70) can be substituted into Eqs. (9.66)–(9.69) to write the desired optical field properties in terms of the phase-screen r_0 values. This substitution yields

$$r_{0,pw} = \left(\sum_{i=1}^{n} r_{0_i}^{-5/3}\right)^{-3/5} \tag{9.71}$$

$$r_{0,sw} = \left[\sum_{i=1}^{n} r_{0_i}^{-5/3}\left(\frac{z_i}{\Delta z}\right)^{5/3}\right]^{-3/5} \tag{9.72}$$

$$\sigma^2_{\chi,pw} = 1.33\,k^{-5/6}\Delta z^{5/6}\sum_{i=1}^{n} r_{0_i}^{-5/3}\left(1-\frac{z_i}{\Delta z}\right)^{5/6} \tag{9.73}$$

$$\sigma^2_{\chi,sw} = 1.33\,k^{-5/6}\Delta z^{5/6}\sum_{i=1}^{n} r_{0_i}^{-5/3}\left(\frac{z_i}{\Delta z}\right)^{5/6}\left(1-\frac{z_i}{\Delta z}\right)^{5/6}. \tag{9.74}$$

Given a set of desired atmospheric conditions, $r_{0,sw}$ and $\sigma^2_{\chi,sw}$ for example, these equations could be used to determine the required phase screen properties and locations along the path. These equations could be written in matrix-vector notation. Using a typical number of phase screens, like 5–10, there are 10–20 unknown parameters (r_0 and z_i for each screen), and so the system of two equations is far underdetermined. This is easy to improve by simply fixing phase screen locations. For example, we could maintain consistency with the uniform spacing of the partial-propagation planes, as discussed in Ch. 8. Then, choosing to place a phase screen in each partial-propagation plane, we can recall from Sec. 8.3 that $\alpha_i = z_i/\Delta z$, which simplifies the equations further. As an example, the system of equations for five screens would look like

$$\begin{pmatrix} r_{0,sw}^{-5/3} \\ \frac{\sigma^2_{\chi,sw}}{1.33}\left(\frac{k}{\Delta z}\right)^{5/6} \end{pmatrix} = \begin{pmatrix} 0 & 0.0992 & 0.315 & 0.619 & 1 \\ 0 & 0.248 & 0.315 & 0.248 & 0 \end{pmatrix} \begin{pmatrix} r_{01}^{-5/3} \\ r_{02}^{-5/3} \\ r_{03}^{-5/3} \\ r_{04}^{-5/3} \\ r_{05}^{-5/3} \end{pmatrix}. \tag{9.75}$$

The entries in the first row of the matrix are $\alpha_i^{5/3}$, and the entries in the second row of the matrix are $\alpha_i^{5/6}\left(1-\alpha_i\right)^{5/6}$.

In this approach, the left side is determined by the scenario we want to simulate. Given λ, Δz, and a model of $C_n^2\left(z\right)$, we compute the desired atmospheric parameters for the simulation. Then, we solve an appropriate system of equations, like Eq. (9.75), to compute the phase screen r_0 values. The difficulty with this approach is the $-5/3$ power in the r_0 vector. Negative entries in the solved r_0 vector are unphysical, so the solutions must be constrained to positive values. The example in Sec. 9.5 shows use of constrained optimization to compute r_0 values for a simulation with several phase screens.

9.3 Monte-Carlo Phase Screens

The refractive index variation of the atmosphere is a random process, and so is the optical path length through it. Consequently, turbulence models give statistical averages, like the structure function and power spectrum of refractive index variations. The problem of creating atmospheric phase screens is one of generating individual realizations of a random process. That is, phase screens are created by transforming computer-generated random numbers into two-dimensional arrays of phase values on a grid of sample points that have the same statistics as turbulence-induced phase variations. The literature is rife with clever methods to generate atmospheric phase screens with good computational efficiency,[72–75] high accuracy,[56,71,76–82] and flexibility.[83–85]

Usually, the phase is written as a weighted sum of basis functions. The common basis sets used for this purpose have been Zernike polynomials and Fourier series. Both basis sets have benefits and drawbacks. The most common method for phase-screen generation is based on the FT, first introduced by McGlamery.[86]

Assuming that turbulence-induced phase $\phi\left(x,y\right)$ is a Fourier-transformable function, we can write it in a Fourier-integral representation as

$$\phi\left(x,y\right)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\Psi\left(f_x,f_y\right)e^{i2\pi\left(f_x x+f_y y\right)}\,df_x\,df_y, \tag{9.76}$$

where $\Psi\left(f_x,f_y\right)$ is the spatial-frequency-domain representation of the phase. Of course, $\phi\left(x,y\right)$ is actually a realization of a random process with a power spectral density given by $\Phi_\phi\left(f\right)$ [or equivalently, $\Phi_\phi\left(\kappa\right)$] as discussed in Sec. 9.2.3. Treating the phase as a two-dimensional signal, the total power P_{tot} in the phase can be written two ways using the definition of power spectral density and Parseval's theorem so that

$$P_{tot}=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left|\phi\left(x,y\right)\right|^2\,dx\,dy=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\Phi_\phi\left(f_x,f_y\right)\,df_x\,df_y. \tag{9.77}$$

To generate phase screens on a finite grid, we write the optical phase $\phi(x, y)$ as a Fourier series so that[80]

$$\phi(x, y) = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} c_{n,m} \exp\left[i2\pi \left(f_{x_n} x + f_{y_m} y\right)\right], \tag{9.78}$$

where f_{x_n} and f_{y_m} are the discrete x- and y-directed spatial frequencies, and the $c_{n,m}$ are the Fourier-series coefficients. Because the phase variation through the atmosphere is due to many independent random inhomogeneities along the optical path, we use the central-limit theorem to determine that the $c_{n,m}$ have a Gaussian distribution. Also note that, in general, the Fourier coefficients $c_{n,m}$ are complex. The real and imaginary parts each have zero mean and equal variances, and their cross-covariances are zero. Consequently, they obey circular complex Gaussian statistics with zero mean and variance given by[32,80]

$$\left\langle \left|c_{n,m}\right|^2 \right\rangle = \Phi_\phi\left(f_{x_n}, f_{y_m}\right) \Delta f_{x_n} \Delta f_{y_m}. \tag{9.79}$$

If the FFT is to be used for computational efficiency, the frequency samples must be linearly spaced on a Cartesian grid. Then, if the x and y grid sizes are L_x and L_y, respectively, the frequency spacings are $\Delta f_{x_n} = 1/L_x$ and $\Delta f_{y_m} = 1/L_y$ so that

$$\left\langle \left|c_{n,m}\right|^2 \right\rangle = \frac{1}{L_x L_y} \Phi_\phi\left(f_{x_n}, f_{y_m}\right). \tag{9.80}$$

Now, the task is to produce realizations of the Fourier coefficients. Typical random-number software, like MATLAB's `randn` function, generates Gaussian random numbers with zero mean and unit variance. This just requires a simple transformation. If x is a Gaussian random variable with mean μ and variance σ^2, then the variable $z = (x - \mu)/\sigma$ is a Gaussian random variable with zero mean and unit variance. With this in mind, we simply generate Gaussian random numbers via standard mathematical software with zero mean and unit variance. Then, multiplication by the square root of the variance given in Eq. (9.79) produces the random draws of the FS coefficients in Eq. (9.78).

Listing 9.2 gives MATLAB code for generating phase screens using the FT method. Lines 6–16 set up the square root of Eq. (9.51). As part of the process, line 16 sets the zero-frequency component of the phase to zero. Then, line 18 generates a random draw of the FS coefficients. Finally, line 20 synthesizes the phase screen from random draws using an FT. Note that the real and imaginary parts of the IFT produce two uncorrelated phase screens. Line 20 uses the screen from the real part and discards the imaginary part.

Unfortunately, the FFT method shown in Listing 9.2 does not produce accurate phase screens. To begin understanding this, the reader should note that the phase PSDs shown in Fig. 9.1 given in Eq. (9.51) have much of the power in the low spatial frequencies. In fact, it has been well documented that we often cannot sample the spatial-frequency grid low enough to accurately represent low-order

Listing 9.2 MATLAB code for generating phase screens that are consistent with atmospheric turbulence from random draws. This code uses the FT method.

```
function phz = ft_phase_screen(r0, N, delta, L0, l0)
% function phz ...
%     = ft_phase_screen(r0, N, delta, L0, l0)

    % setup the PSD
    del_f = 1/(N*delta);   % frequency grid spacing [1/m]
    fx = (-N/2 : N/2-1) * del_f;
    % frequency grid [1/m]
    [fx fy] = meshgrid(fx);
    [th f] = cart2pol(fx, fy);  % polar grid
    fm = 5.92/l0/(2*pi); % inner scale frequency [1/m]
    f0 = 1/L0;           % outer scale frequency [1/m]
    % modified von Karman atmospheric phase PSD
    PSD_phi = 0.023*r0^(-5/3) * exp(-(f/fm).^2) ...
        ./ (f.^2 + f0^2).^(11/6);
    PSD_phi(N/2+1,N/2+1) = 0;
    % random draws of Fourier coefficients
    cn = (randn(N) + i*randn(N)) .* sqrt(PSD_phi)*del_f;
    % synthesize the phase screen
    phz = real(ift2(cn, 1));
```

modes like tilt. This difference is evident in Fig. 9.3 when we generate and verify phase screens for an example simulation through turbulence. For this figure, 40 turbulent phase screens were generated using the FT method implemented by the `ft_phase_screen` function in Listing 9.2. Then, the structure function of each screen was computed using the `str_fcn2_ft` function in Listing 3.7, and the results were averaged. A slice of the average structure function is shown by the dotted line. Clearly, the screens' statistics do not match up well with the theoretical structure function shown by the solid gray line. The poorest agreement is at large separations, which correspond to low spatial frequencies.

Several approaches have been suggested to compensate for this shortcoming. For example, Cochran,[76] Roddier,[87] and Jakobssen[79] use random draws of Zernike polynomials (or linear combinations thereof) using the Zernike-mode statistics reported by Noll.[22] In contrast, Welsh[80] and Eckert and Goda[82] use FS methods with non-uniform sampling in the spatial-frequency domain to include very low spatial frequencies. Still others use a combination of these two approaches, called "subharmonics". This approach, used by Herman and Strugala,[77] Lane *et al.*,[78] Johansson and Gavel,[88] and Sedmak,[81] augments FT screens with a low-frequency Fourier series.

Here, we implement the subharmonic method described by Lane *et al.*[78] Frehlich

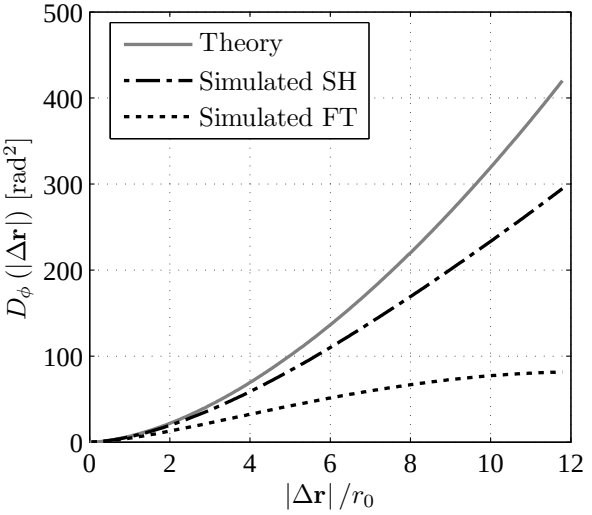

Figure 9.3 Comparison of the average structure function computed from FT and subharmonic screens against theory.

showed that turbulent simulations using these screens produce accurate results.[56] Listing 9.3 gives MATLAB code for generating phase screens using this method. In Line 8, this method begins by generating a phase screen using the FT method already discussed. Then, a low-frequency screen is generated in lines 12–37. This screen $\phi_{LF}(x, y)$ is a sum of N_p different screens, as given by

$$\phi_{LF}(x,y) = \sum_{p=1}^{N_p} \sum_{n=-1}^{1} \sum_{m=-1}^{1} c_{n,m} \exp\left[i2\pi\left(f_{x_n} x + f_{y_m} y\right)\right], \tag{9.81}$$

where the sums over n and m are over discrete frequencies and each value of the index p corresponds to a different grid. The square root of the PSD is setup in lines 14–26, the random draws of Fourier coefficients are generated in lines 28–29, and the sum over the indices n and m is carried out in lines 32–35. Then, the sum over the N_p different grids is carried out in line 36. In this particular implementation, only a 3×3 grid of frequencies is used for each value of p, and $N_p = 3$ different grids are used. The frequency grid spacing for each value of p is $\Delta f_p = 1/(3^p L)$. In this way, the frequency grids have a spacing that is a subharmonic of the FT screen's grid spacing.

Listing 9.4 gives an example of generating random phase screens using the MATLAB function `ft_sh_phase_screen` from Listing 9.3. In the listing, the screen size is 2 m, the coherence diameter is $r_0 = 10$ cm, the inner scale is $l_0 = 1$ cm, and the outer scale is $L_0 = 100$ m. An atmospheric phase-screen realization generated by Listing 9.4 is shown in Fig. 9.4.

Figure 9.3 shows verification that subharmonic screens do produce more-accurate phase screen statistics. Several authors have investigated the subharmonic

Listing 9.3 MATLAB code for generating phase screens that are consistent with atmospheric turbulence from random draws. This code uses the FT method augmented with subharmonics.

```
function [phz_lo phz_hi] ...
    = ft_sh_phase_screen(r0, N, delta, L0, l0)
% function [phz_lo phz_hi] ...
%     = ft_sh_phase_screen(r0, N, delta, L0, l0)

    D = N*delta;
    % high-frequency screen from FFT method
    phz_hi = ft_phase_screen(r0, N, delta, L0, l0);
    % spatial grid [m]
    [x y] = meshgrid((-N/2 : N/2-1) * delta);
    % initialize low-freq screen
    phz_lo = zeros(size(phz_hi));
    % loop over frequency grids with spacing 1/(3^p*L)
    for p = 1:3
        % setup the PSD
        del_f = 1 / (3^p*D); %frequency grid spacing [1/m]
        fx = (-1 : 1) * del_f;
        % frequency grid [1/m]
        [fx fy] = meshgrid(fx);
        [th f] = cart2pol(fx, fy);  % polar grid
        fm = 5.92/l0/(2*pi); % inner scale frequency [1/m]
        f0 = 1/L0;           % outer scale frequency [1/m]
        % modified von Karman atmospheric phase PSD
        PSD_phi = 0.023*r0^(-5/3) * exp(-(f/fm).^2) ...
            ./ (f.^2 + f0^2).^(11/6);
        PSD_phi(2,2) = 0;
        % random draws of Fourier coefficients
        cn = (randn(3) + i*randn(3)) ...
            .* sqrt(PSD_phi)*del_f;
        SH = zeros(N);
        % loop over frequencies on this grid
        for ii = 1:9
            SH = SH + cn(ii) ...
                * exp(i*2*pi*(fx(ii)*x+fy(ii)*y));
        end
        phz_lo = phz_lo + SH;   % accumulate subharmonics
    end
    phz_lo = real(phz_lo) - mean(real(phz_lo(:)));
```

Listing 9.4 Example usage of `ft_sh_phase_screen` function

```
% example_ft_sh_phase_screen.m

D = 2;     % length of one side of square phase screen [m]
r0 = 0.1; % coherence diameter [m]
N = 256;  % number of grid points per side
L0 = 100; % outer scale [m]
l0 = 0.01;% inner scale [m]

delta = D/N;     % grid spacing [m]
% spatial grid
x = (-N/2 : N/2-1) * delta;
y = x;
% generate a random draw of an atmospheric phase screen
[phz_lo phz_hi] ...
     = ft_sh_phase_screen(r0, N, delta, L0, l0);
phz = phz_lo + phz_hi;
```

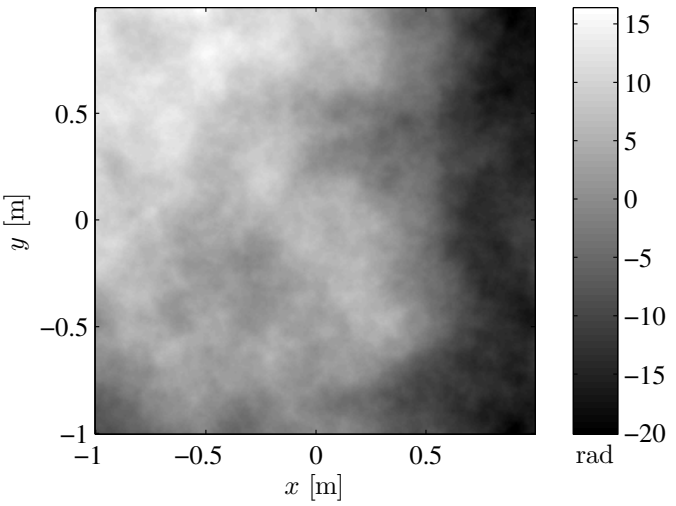

Figure 9.4 Typical atmospheric phase screen created using the subharmonic method.

method's ability to do this. Among the first to do this were Herman and Strugala.[77] While they used a slightly different version of the subharmonic method, they did show that the concept produces phase screens that result in a structure function with a good match to theory. Further, they compared the average Strehl ratio from their subharmonic screens, and it matched theory closely. Later, Lane *et al.* developed the particular subharmonic method used here and demonstrated that their screens also matched the theoretical structure function closely.[78] Shortly thereafter, Johansson and Gavel compared the approaches of Herman and Strugala and Lane *et al.*, and demonstrated their own subharmonic technique whose screens produce a structure function that matches theory very closely.[88] While investigating accuracy of non-square subharmonic phase screens, Sedmak showed good agreement with phase structure function and aperture-averaged phase variance.[81] Finally, Frehlich studied the accuracy of full wave-optics simulations making use of subharmonic screens.[56] His study showed that for beam waves, the mean irradiance is fairly accurate for both FT screens and subharmonic screens, but the subharmonic screens are far more accurate in producing the correct irradiance variance. For plane waves, both methods produced accurate irradiance variances, but only the subharmonic method produced an accurate mutual coherence function.

9.4 Sampling Constraints

As light propagates through turbulence, it spreads due to two effects: tilt and higher-order aberrations. High-order aberrations cause the beam to expand beyond the spreading due to diffraction alone. Tilt causes the beam to wander off the optical axis in a random way. Over time ($\gtrsim$ 1 msec), this random wandering causes optical energy to land all over the observation plane. Beam spreading due to high-order aberrations can be seen in a short-exposure image, whereas beam spreading due to tilt can only be seen in a long-exposure image. A full discussion of beam spreading is beyond the scope of this book, but a simple model for sampling analysis is presented below.

This turbulence-induced beam spreading makes sampling requirements even more restrictive than the vacuum constraints from Sec. 8.4. Several approaches for conducting properly sampled turbulence simulations have been discussed. For example, in vacuum propagation Johnston and Lane filter the free-space transfer function and set their grid size based the bandwidth of the filter.[41] Then, they set the sample interval based on avoiding aliasing of the quadratic phase factor just like in Sec. 7.3.2. For atmospheric simulations, they choose the grid spacing based on the phase structure function. In doing so, they compute the grid spacing δ_ϕ at which phase differences less than π in adjacent grid points occur more than 99.7% of the time. They also give consideration to sampling scintillation. The scale size of scintillation is given approximately by the Fresnel length $(\lambda \Delta z)^{1/2}$, so they set δ_i to be the smallest of δ_ϕ, $(\lambda \Delta z)^{1/2}/2$, and the grid spacing that just barely avoids aliasing of the free-space point spread function. In this way, they adequately sample

free-space propagation and turbulent phase and amplitude variations. Martin and Flatté studied sampling constraints, mainly based on the PSD of the turbulence-induced irradiance fluctuations.[43] Finally, Coles *et al.* conducted a quantitative error analysis for plane waves and point sources.[32] In particular, they studied the error in observation-plane irradiance due to finite grid spacing, finite number of samples, and finite number of screens. They used only FT phase screens, so part of the error they encountered was due to the screens themselves.

Mansell, Praus, and Coy take a different approach, but one that integrates well with the frameworks presented in Chs. 7–8.[35,42,54] They modify the sampling inequalities to account for turbulence-induced beam spreading. The two sampling constraints that originate from propagation geometry are affected by turbulence. The other constraint that originates from the numerical algorithm is not affected by turbulence.

Previously, constraints 1 and 2 were stated for vacuum propagation as

$$1. \qquad \delta_n \leq \frac{\lambda \Delta z - D_2 \delta_1}{D_1} \tag{9.82}$$

$$2. \qquad N \geq \frac{D_1}{2\delta_1} + \frac{D_2}{2\delta_n} + \frac{\lambda \Delta z}{2\delta_1 \delta_n}. \tag{9.83}$$

Constraint 1 ensures that the source-plane grid is sampled finely enough so that all of the rays that land within the observation-plane region of interest are present in the source. In the geometric-optics approximation, turbulence causes the source's rays to refract randomly as shown in Fig. 9.5. This blurs the size of D_1 as viewed in the observation plane and the size of D_2 as viewed in the source plane. We need a model for this blurring that depends on the turbulence to adjust these two constraints.

The approach of Coy is to model the turbulence-induced beam spreading as if it were caused by a diffraction grating with period equal to r_0. This allows us to define new limiting aperture sizes D_1' and D_2' via

$$D_1' = D_1 + c\frac{\lambda \Delta z}{r_{0,rev}} \tag{9.84}$$

$$D_2' = D_2 + c\frac{\lambda \Delta z}{r_0}, \tag{9.85}$$

where $r_{0,rev}$ is the coherence diameter computed for light propagating in reverse, i.e., from the observation plane to the source plane, and c is an adjustable parameter indicating the sensitivity of the model to the turbulence. Typical values of c range from 2 to 8. Choosing $c = 2$ typically captures $\sim$97% of the light, and choosing $c = 4$ typically captures $\sim$99% of the light. Now, for simulating propagation through turbulence, the required sampling analysis utilizes the following inequalities:

$$1. \quad \delta_n \leq \frac{\lambda \Delta z - D_2' \delta_1}{D_1'} \tag{9.86}$$

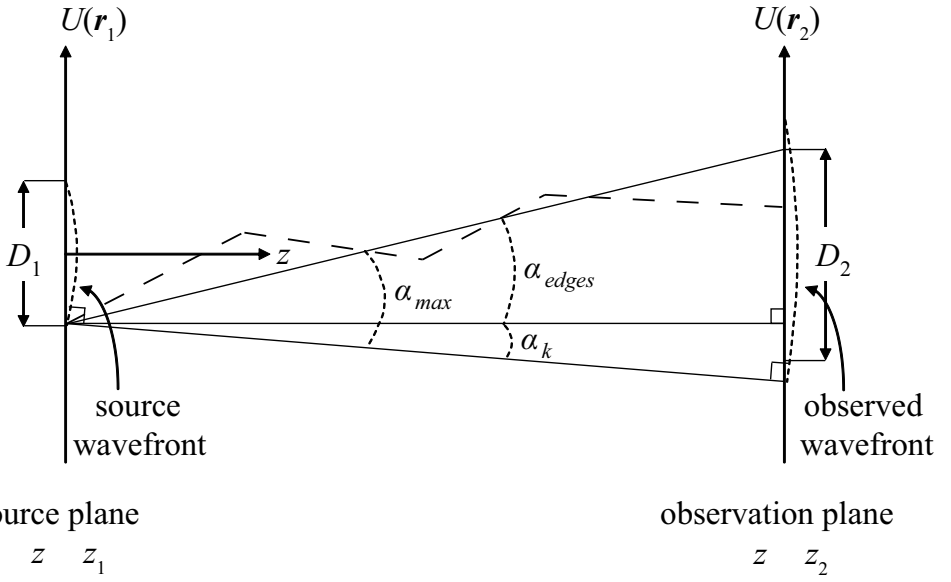

Figure 9.5 Propagation geometry in which turbulence refracts rays as the light propagates indicated by the dashed ray. This geometry leads to Constraint 1.

$$2.\quad N \geq \frac{D_1'}{2\delta_1} + \frac{D_2'}{2\delta_n} + \frac{\lambda \Delta z}{2\delta_1 \delta_n} \tag{9.87}$$

$$3.\quad \left(1 + \frac{\Delta z}{R}\right)\delta_1 - \frac{\lambda \Delta z}{D_1} \leq \delta_2 \leq \left(1 + \frac{\Delta z}{R}\right)\delta_1 + \frac{\lambda \Delta z}{D_1}. \tag{9.88}$$

Then, once N, δ_1, and δ_n are chosen, the partial propagation distances and number of partial propagations are chosen from

$$\Delta z_{max} = \frac{\min\left(\delta_1, \delta_n\right)^2 N}{\lambda} \tag{9.89}$$

$$n_{min} = \text{ceil}\left(\frac{\Delta z}{\Delta z_{max}}\right) + 1, \tag{9.90}$$

as before.

9.5 Executing a Properly Sampled Simulation

As in Chs. 7–8, the most effective way to illustrate application of the above sampling constraints is by example. The remainder of this section illustrates the steps involved in setting up a simulation of optical-wave propagation through atmospheric turbulence.

9.5.1 Determine propagation geometry and turbulence conditions

The example simulation in this subsection is for a point source propagating a distance $\Delta z = 50$ km through a turbulent path with $C_n^2 = 1 \times 10^{-16}$ m$^{-2/3}$ along the

Listing 9.5 MATLAB code for setting up source and receiver geometry and turbulence-related quantities.

```
% example_pt_source_atmos_setup.m

% determine geometry
D2 = 0.5;   % diameter of the observation aperture [m]
wvl = 1e-6;  % optical wavelength [m]
k = 2*pi / wvl; % optical wavenumber [rad/m]
Dz = 50e3;     % propagation distance [m]

% use sinc to model pt source
DROI = 4 * D2;  % diam of obs-plane region of interest [m]
D1 = wvl*Dz / DROI;    % width of central lobe [m]
R = Dz; % wavefront radius of curvature [m]

% atmospheric properties
Cn2 = 1e-16;    % structure parameter [m^-2/3], constant
% SW and PW coherence diameters [m]
r0sw = (0.423 * k^2 * Cn2 * 3/8 * Dz)^(-3/5);
r0pw = (0.423 * k^2 * Cn2 * Dz)^(-3/5);
p = linspace(0, Dz, 1e3);
% log-amplitude variance
rytov = 0.563 * k^(7/6) * sum(Cn2 * (1-p/Dz).^(5/6) ...
    .* p.^(5/6) * (p(2)-p(1)));

% screen properties
nscr = 11; % number of screens
A = zeros(2, nscr); % matrix
alpha = (0:nscr-1) / (nscr-1);
A(1,:) = alpha.^(5/3);
A(2,:) = (1 - alpha).^(5/6) .* alpha.^(5/6);
b = [r0sw.^(-5/3); rytov/1.33*(k/Dz)^(5/6)];
% initial guess
x0 = (nscr/3*r0sw * ones(nscr, 1)).^(-5/3);
% objective function
fun = @(X) sum((A*X(:) - b).^2);
% constraints
x1 = zeros(nscr, 1);
rmax = 0.1; % maximum Rytov number per partial prop
x2 = rmax/1.33*(k/Dz)^(5/6) ./ A(2,:);
x2(A(2,:)==0) = 50^(-5/3)
[X,fval,exitflag,output] ...
    = fmincon(fun,x0,[],[],[],[],x1,x2)
% check screen r0s
r0scrn = X.^(-3/5)
r0scrn(isinf(r0scrn)) = 1e6;
% check resulting r0sw & rytov
bp = A*X(:); [bp(1)^(-3/5) bp(2)*1.33*(Dz/k)^(5/6)]
[r0sw rytov]
```

entire path. For simplicity, we assume that the Kolmogorov refractive-index PSD is adequate for our purposes. The telescope observing the light is $D_2 = 0.5$ m in diameter. With this information, we can compute the atmospheric parameters of interest. This, of course, depends on what we want to do with the light after propagation. Perhaps we may want to do imaging, wavefront sensing, adaptive optics, and more. In this particular example, we are simply interested in verifying that the simulation is operating correctly. To verify, we propagate the source through many realizations of turbulence, compute the coherence factor, and plot it against the theoretical expectation. We also need to determine the locations of the phase screens and their coherence diameters.

Listing 9.5 gives the MATLAB code for setting up the turbulence model. This starts with setting aperture sizes, optical wavelength, propagation distance, etc. Lines 10–11 compute D_1 from the width of the model point source's central lobe. This begins with setting the diameter of the region of interest (the variable `DROI`) that is uniformly illuminated in the observation plane by the source. Lines 17–22 continue with computing the key atmospheric parameters, $r_{0,sw} = 12.7$ cm and $\sigma^2_{\chi,sw} = 0.436$, from Eqs. (9.43) and (9.64), respectively.

Lines 25–41 compute the phase screen r_0 values according to the approach in Sec. 9.2.5. In this process, lines 26–29 set up the matrix, which is similar to the matrix in Eq. (9.75). Line 30 sets up the vector, which is the left side in Eq. (9.75) (the variable `b`). With the known matrix and vector determined, the screen r_0 values must be computed through a constrained search through possible values of screen r_0's. Actually, the parameters are the $-5/3$ power of the screen r_0's in the variable `X` according to Eq. (9.75). Their values are computed through constructing an objective function that can be minimized when suitable r_0 values are found within a valid range. This objective function in line 34 is the difference between the desired atmospheric parameters (the variable `b`) and those arising from a given choice of r_0 values (`A*X(:)`). The valid range of the `X` values is determined in lines 36–39. The lower bound of `X` is zero, corresponding to infinite screen r_0's. The upper bound is set by requiring that each screen's contribution to the overall Rytov number is less than 0.1 (see line 37). This is related to a guideline suggested by Martin and Flatté.[43] Finally, lines 40–41 perform the search to minimize the objective function, and lines 46–47 compute the atmospheric parameters based on the solved screen r_0's and print them to the command line.

9.5.2 Analyze the sampling constraints

Once the geometry and turbulence conditions are set up, we can analyze the sampling constraints to determine the grid spacings and number of grid points. In Listing 9.6, we evaluate Eqs. (9.86)–(9.88) and perform a sampling analysis using essentially the same method as in Sec. 8.4. Lines 2–16 evaluate the bounds of constraints 1–3. This is used to produce the contour plot shown in Fig. 9.6, although the plotting code is not shown. The figure shows the lower bound on N in constraint 2

Listing 9.6 MATLAB code for analyzing sampling constraints given the geometry and turbulence conditions.

```
% analysis_pt_source_atmos_samp.m
c = 2;
D1p = D1 + c*wvl*Dz/r0sw;
D2p = D2 + c*wvl*Dz/r0sw;

delta1 = linspace(0, 1.1*wvl*Dz/D2p, 100);
deltan = linspace(0, 1.1*wvl*Dz/D1p, 100);
% constraint 1
deltan_max = -D2p/D1p*delta1 + wvl*Dz/D1p;
% constraint 3
d2min3 = (1+Dz/R)*delta1 - wvl*Dz/D1p;
d2max3 = (1+Dz/R)*delta1 + wvl*Dz/D1p;
[delta1 deltan] = meshgrid(delta1, deltan);
% constraint 2
N2 = (wvl * Dz + D1p*deltan + D2p*delta1) ...
    ./ (2 * delta1 .* deltan);
% constraint 4
d1 = 10e-3;
d2 = 10e-3;
N = 512;
d1*d2 * N / wvl
zmax = min([d1 d2])^2 * N / wvl
nmin = ceil(Dz / zmax) + 1
```

with the upper bound from constraints 1 and 3 overlayed. This allows us to choose the grid spacings δ_1 and δ_n in the source and observation planes, respectively, and the minimum required number of grid points, N. Then, given our choices for δ_1, δ_n, and N, we can compute the maximum allowed propagation distance Δz_{max} using Eq. (9.89) and then corresponding number of partial propagations, $n-1$, using Eq. (9.90).

The results of the analysis are given in lines 18–23, which assumes that we have already made the plots and viewed them. The chosen grid spacings are $\delta_1 = 1$ cm, and $\delta_n = 1$ cm. This gives five samples across the central peak of the model point source and 50 samples across the observing telescope aperture. This is marked on Fig. 9.6 with a white $\times$. We can see that these spacings easily satisfy constraints 1 and 3. Also, the required number of grid points is more than 2^8, so we pick $2^9 = 512$ grid points. Finally, the minimum number of planes is two, so we could use just one propagation. However, we use ten propagations (11 planes) to represent the atmosphere properly.

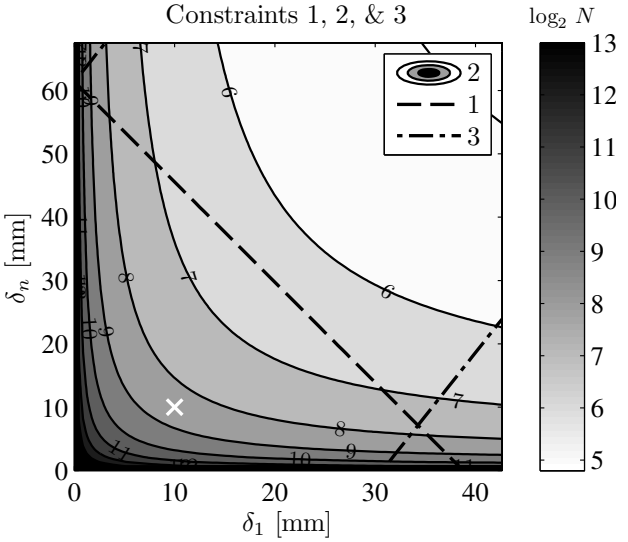

Figure 9.6 Graphical sampling analysis for the example point-source propagation. The region that satisfies constraint 1 is below the black dashed line, while the region above the black dash-dot line satisfies constraint 3. The white $\times$ marks the chosen values of δ_1 and δ_n.

9.5.3 Perform a vacuum simulation

With the grid parameters N, δ_1, and δ_n determined, the next step is to perform a vacuum simulation. This serves two important purposes. First, it verifies that the simulation is producing accurate results without regard to the turbulence. In this particular case, we are simulating a point source, so we can compare the vacuum simulation result against a known analytic solution. Listing 9.7 gives the MATLAB code that carries out a vacuum simulation for the example geometry. Lines 3–5 create copies of some variables from Listing 9.6. Then, lines 12–14 create the sinc-Gaussian model point source. Next, lines 19–25 setup and perform the propagation using a super-Gaussian absorbing boundary at each plane. Lastly, the computed field is collimated by removing the spherical-wave phase. This allows the phase difference to be studied, which is helpful for making some plots and absolutely necessary for certain analyses, like computing the coherence factor.

False-color, gray-scale images of the resulting irradiance and phase are shown in Fig. 9.7. Clearly, the irradiance in plot (a) is nearly uniform over the region of interest, and the phase in plot (b) is flat (after collimation). Plotting a slice of the phase with the theoretical expectation would reveal that the curvature is correct. The second purpose of performing a vacuum simulation is for comparison to the turbulent simulations. Often, we want to know how much the performance of an optical system is degraded by turbulence, so we need to know how the system performs in vacuum for comparison. This is necessary, for example, if we want to calculate the Strehl ratio.

Listing 9.7 MATLAB code for executing a vacuum simulation of the point source given the grid determined by sampling analysis.

```
% example_pt_source_vac_prop.m

delta1 = d1;     % source-plane grid spacing [m]
deltan = d2;     % observation-plane grid spacing [m]
n = nscr;          % number of planes

% coordinates
[x1 y1] = meshgrid((-N/2 : N/2-1) * delta1);
[theta1 r1] = cart2pol(x1, y1);

% point source
pt = exp(-i*k/(2*R) * r1.^2) / D1^2 ...
    .* sinc(x1/D1) .* sinc(y1/D1) ...
    .* exp(-(r1/(4*D1)).^2);
% partial prop planes
z = (1 : n-1) * Dz / (n-1);

% simulate vacuum propagation
sg = exp(-(x1/(0.47*N*d1)).^16) ...
    .* exp(-(y1/(0.47*N*d1)).^16);
t = repmat(sg, [1 1 n]);
[xn yn Uvac] = ang_spec_multi_prop(pt, wvl, ...
    delta1, deltan, z, t);
% collimate the beam
Uvac = Uvac .* exp(-i*pi/(wvl*R)*(xn.^2+yn.^2));
```

9.5.4 Perform the turbulent simulations

Finally, we can perform turbulent simulations with realizations of phase screens. Listing 9.8 gives the code for executing turbulent simulations for the example scenario. In the listing, we generate 11 phase screens (at the correct grid spacings, which may be different for each screen) to create one realization of a turbulent path and simulate the propagation. The process is repeated 40 times so that we have 40 realizations of optical fields propagated through independent and identically distributed atmospheres. A false-color, gray-scale image of one representative field is shown in Fig. 9.8 with the irradiance in plot (a) and phase in plot (b). Collecting many such realizations allows us to estimate ensemble statistics like the coherence factor, wave structure function, and log-amplitude variance.

If we wanted to simulate a dynamically evolving atmosphere, for each atmospheric realization we would need to move the phase screens in the transverse dimension as time evolves. This makes explicit use of the Taylor frozen-turbulence

hypothesis.[15] The velocities of the screens needs to be determined from temporal quantities like the Greenwood frequency.[68] This would allow us to verify temporal properties of the simulation and then use the simulation with dynamic optical systems such as adaptive optics.

9.5.5 Verify the output

There are two simulation properties that are verified in this subsection. The first is the phase-screen structure function, and the second is the coherence factor of the observation-plane field. These verifications make use of independent and identically distributed realizations to check spatial correlations. If a dynamically evolving atmosphere is simulated, temporal properties like the temporal phase structure function should be checked as well.

First, the phase screens are verified. To do so, we can use the 40 random draws for any one partial-propagation plane. This is done by computing the two-

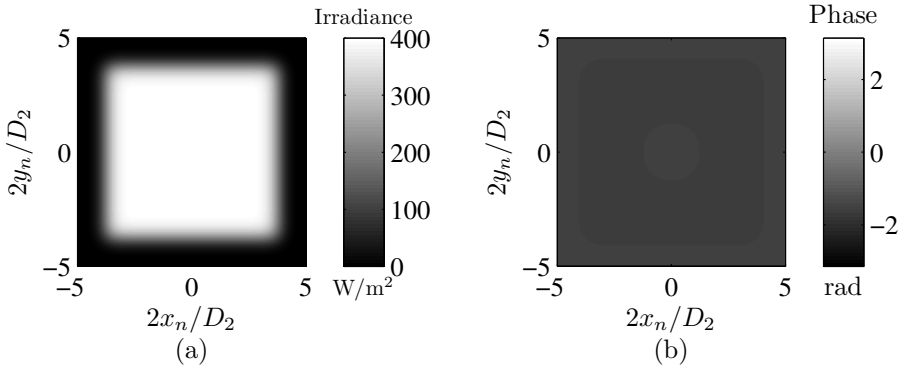

Figure 9.7 Irradiance and phase resulting from a vacuum propagation of the model point source. Note that line 25 of Listing 9.7 indicates that the field was collimated before plotting, which is visible in plot (b).

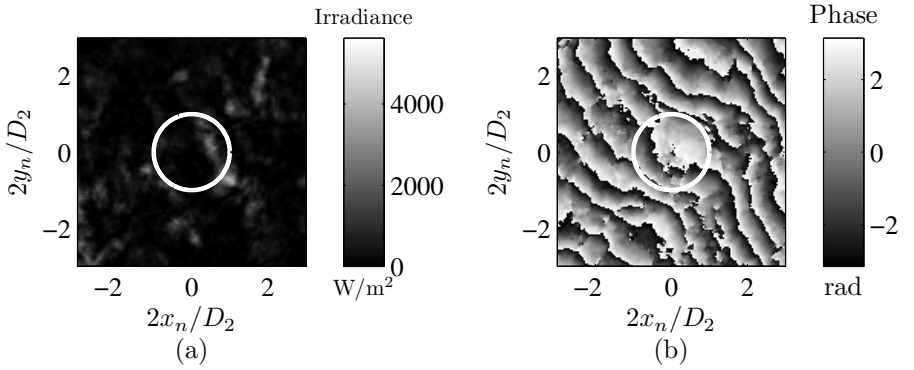

Figure 9.8 Irradiance (a) and phase (b) resulting from a turbulent propagation of the model point source. The white circle marks the edge of the observing telescope aperture. Note that the field was collimated before plotting, which is apparent in plot (b).

Listing 9.8 MATLAB code for executing a turbulent simulation of the point source given the grid determined by sampling analysis.

```
% example_pt_source_turb_prop.m

l0 = 0;      % inner scale [m]
L0 = inf;    % outer scale [m]

zt = [0 z];  % propagation plane locations
Delta_z = zt(2:n) - zt(1:n-1);    % propagation distances
% grid spacings
alpha = zt / zt(n);
delta = (1-alpha) * delta1 + alpha * deltan;

% initialize array for phase screens
phz = zeros(N, N, n);
nreals = 20;    % number of random realizations
% initialize arrays for propagated fields,
% aperture mask, and MCF
Uout = zeros(N);
mask = circ(xn/D2, yn/D2, 1);
MCF2 = zeros(N);
sg = repmat(sg, [1 1 n]);
for idxreal = 1 : nreals     % loop over realizations
    idxreal
    % loop over screens
    for idxscr = 1 : 1 : n
        [phz_lo phz_hi] ...
            = ft_sh_phase_screen ...
            (r0scrn(idxscr), N, delta(idxscr), L0, l0);
        phz(:,:,idxscr) = phz_lo + phz_hi;
    end
    % simulate turbulent propagation
    [xn yn Uout] = ang_spec_multi_prop(pt, wvl, ....
        delta1, deltan, z, sg.*exp(i*phz));
    % collimate the beam
    Uout = Uout .* exp(-i*pi/(wvl*R)*(xn.^2+yn.^2));
    % accumulate realizations of the MCF
    MCF2 = MCF2 + corr2_ft(Uout, Uout, mask, deltan);
end
% modulus of the complex degree of coherence
MCDOC2 = abs(MCF2) / (MCF2(N/2+1,N/2+1));
```

dimensional structure function of each phase screen and then averaging each to obtain the mean structure function, as discussed in Sec. 3.3. Figure 9.9 shows an example comparison of the theoretical phase structure function from Eq. (9.44) to

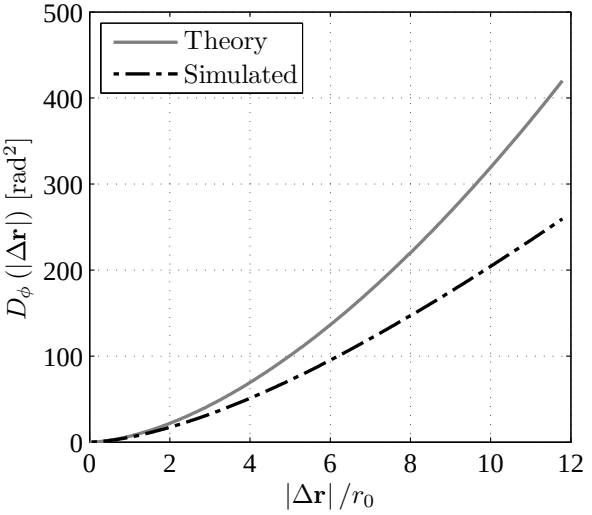

Figure 9.9 Verifying structure function of an ensemble of independent and identically distributed phase screens.

the average structure function computed from phase screen realizations. The comparison is close, indicating that the screens are adequately representing the phase accumulated along the propagation path.

To confirm that the turbulent simulation operates correctly, we have computed the coherence factor in the observation plane. Line 36 in Listing 9.8 accumulates the two-dimensional mutual coherence function using the `corr2_ft` function from Ch. 3, and line 39 normalizes to get the coherence factor. The result is plotted in Fig. 9.10 along with the theoretical expectation. The theoretical expectation combines Eqs. (9.32) and (9.44). We can see that there is a good match between theory and the simulation results. There is a slight departure, so if we need greater accuracy, we could go back to the setup and re-evaluate the choice of phase screen properties ro try an even more accurate screen generation method like the one developed by Johansson and Gavel.[88] One way to adjust the setup would be to examine Eq. (9.65) and adjust the values of z_i and Δz_i attempting to match turbulence moments of the continuous and layered models. The case of constant C_n^2 discussed here is a simple case for which uniformly spaced screens with uniform properties work fairly well. As an example of more extensive verification that could be performed, Martin and Flatté[43,44] tested their simulations by comparing the spatial irradiance PSD in the observation plane against weak turbulence theory and asymptotic theory.

9.6 Conclusion

The example given in this chapter has illustrated the steps that we must take to set up a simulation of optical propagation through turbulence and ensure accurate results. This is an important process, and many of these steps are often overlooked.

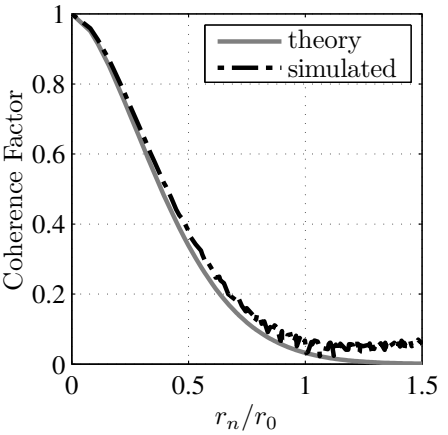

Figure 9.10 The coherence factor in the observation plane.

Because simulations can be much more complicated than the situation given here, often more effort is required to ensure accurate simulation results. Additional complexities often include two-way propagation, adaptive-optics systems, moving platforms, reflection from rough surfaces, multiple wavelengths, and much more.[36,89] These additions need to be tested as thoroughly as the atmospheric propagation part of the simulation.

9.7 Problems

1. Show that if ϵ is position-dependent, Maxwell's equations combine similarly to the development in Sec. 1.2.1 to yield Eq. (9.20).

2. Show that for a propagation path with constant C_n^2, $r_{0,sw} = (3/8)^{-3/5} r_{0,pw}$.

3. Substitute Eq. (9.44) into Eq. (9.34) to show that Eq. (9.49) is the correct phase PSD for Kolmogorov turbulence.

4. Show that for a propagation path with constant C_n^2, $\sigma^2_{\chi,sw} = 0.404\sigma^2_{\chi,pw}$.

5. Show the sampling diagram for a point source with wavelength 1 μm propagating 2 km through an atmosphere with $r_0 = 2$ cm to a telescope with a 2-m-diameter aperture. Compare this to the vacuum case. How many more samples are needed? How many partial propagations are needed in each case?

6. Show the sampling diagram for a point source with wavelength 1 μm propagating 75 km through an atmosphere with $r_0 = 10$ cm to a telescope with a 1-m-diameter aperture. Compare this to the vacuum case. How many more samples are needed? How many partial propagations are needed in each case?

7. Consider propagating a point source with an optical wavelength of 1 μm a distance $\Delta z = 100$ km through an atmosphere with the Kolmogorov refractive-index PSD and $C_n^2 = 1 \times 10^{-17}$ m$^{-2/3}$ all along the path.

   (a) Analytically evaluate the integrals given in Eqs. (9.42), (9.43), (9.63), and (9.64) to compute the continuous-model r_0 and log-amplitude variance σ_χ^2 for both a plane wave and a point source, assuming that C_n^2 is constant along the propagation path.

   (b) Using three phase screens, write down the matrix-vector equations similar to Eq. (9.75) in an attempt to match the continuous and discrete point-source r_0, point-source log-amplitude variance, and plane-wave log-amplitude variance. Solve the system of equations for the three values of r_{0i}. With three parameters and three screens, there is a unique solution. Is it physically meaningful? Explain your answer.

   (c) Now, adapt the system of equations to accommodate seven phase screens and solve the system similarly to the method in Listing 9.5.

   (d) Given that the receiving aperture has a diameter of 2 m, perform the sampling analysis with consideration of the turbulence. Create a plot similar to Fig. 8.5

   (e) Generate the phase screens with 20 independent and identically distributed realizations using the Kolmogorov phase PSD. Compute the structure function for the last phase screen and plot it along with the appropriate theoretical expectation.

   (f) Simulate the propagation through the turbulent path and plot the coherence factor of the observation-plane field along with the theoretical expectation.