Physics and Astronomy, School of Physical and Chemical Sciences,
University of Canterbury, Christchurch, New Zealand

Visualising the Freudenreich 1998 Model of the Galactic Bar and Disk

Joseph Wilson

PHYS391 Project 2018

Supervisor: C. Gordon

Abstract

The model of the galactic bar and disk proposed by Freudenreich (1998) [3] is an empirical model which was derived from the survey of the Diffuse Infrared Background Experiment of the *Cosmic Background Explorer* (conducted in 1989–1990). In this report, Freudenreich's model is employed to produce an intensity sky map. Important details or concepts involved in the process of rendering the model are elaborated on, wherever Freudenreich gives little explanation. The coordinate systems are defined, photometric concepts are clarified, the method for computing the model is given, and errors or ambiguities in Freudenreich (1998) are corrected.

Contents

1 Introduction

Many investigations into the structure of the galaxy focus on counting stars (or related objects) as a function of position and distance, to create a three-dimensional map from which a model may be derived. Two difficulties with this straightforward method are accurately determining the distances of objects and obtaining samples in an unbiased nature. Examples of studies based on this star-counting method include Bahcall (1986) [1] and, more recently, Pandey, Sharma & Ogura (2006) [5]—but star-counting is not the only method in use.

Freudenreich (1998) [3] applies photometry to study the galaxy; a technique which has the advantage of having greater range than star-counting. (This is because as distance D increases, the dimming of sources $\propto D^{-2}$ is compensated for by the increasing area subtended by a solid angle $\propto D^2$, ignoring extinction.) The *Cosmic Background Explorer* produced a full-sky data set at infrared wavelengths during the Diffuse Infrared Background Experiment (DIRBE) conducted in 1989–1990, which contained maps at wavelengths ranging from 1.25 µm to 240 µm. Freudenreich uses the near-infrared 1.25 µm, 2.2 µm, 3.5 µm and 4.9 µm bands (within the standard photometric bands J, K, L and M, respectively) to study the galactic background. These maps are used because they do not suffer from extinction nearly as much as visible wavelength surveys. They are also a good choice for photometric analysis of the galaxy because the near-infrared maps are dominated by red giants, whose uniform abundance makes them well-representative of the structure of the disk [2, §1].[1] Freudenreich proposes three closely-related parametric models of the galactic bar and disk derived from these DIRBE maps (named the S, E and P models, after their defining bulge shape).

This report describes the process of employing Freudenreich's model to produce a celestial intensity map which can be compared to DIRBE observations. Important details in this process which are not given much explanation by Freudenreich are elaborated upon, and mathematical or typographical errors in Freudenreich (1998) are identified and corrected. The rest of the introduction defines coordinates, summarises how the DIRBE data were prepared, and gives an overview of the photometric concepts involved. Section 2 gives Freudenreich's parametric model in detail, and section 3 describes how the model was used to produce an image, which is then compared to observed DIRBE data.

[1] Since this report makes frequent reference to Freudenreich, citations also contain a section number when needed. For instance, the last citation—[2, §1]—refers specifically to section 1 of [2].

1.1 Coordinates

Heliocentric spherical galactic coordinates (l, b, s) are used to map space from the perspective of the sun. The sun is defined to be at the origin with $(l, b, s) = (0°, 0°, 0\,\mathrm{kpc})$, and the line $(l, b) = (0°, 0°)$ is directed toward the centre of the galaxy. Galactic latitude l and longitude b measure angular distance westward and northward from the galactic centre, respectively, where *northward* is perpendicular to the galactic equator, on the side to which Earth's north pole points.

The model itself employs Cartesian and cylindrical galactocentric coordinates, in which the centre of the galaxy is defined as the origin. The sun is located at $(X, Y, Z) = (-R_\odot, 0, Z_\odot)$ or $(R, \theta, Z) = (R_\odot, 180°, Z_\odot)$ where $R_\odot$ is the distance from the galactic centre to the point in the galactic plane closest to the sun, $Z_\odot$ is the height of the sun above the galactic plane, and θ is measured counter-clockwise looking down from $+Z$.

The model also makes use of coordinates in the frame of the galactic bar, which is concentric to the galactic frame, but rotated with a pitch angle φ_{bar} and yaw θ_{bar} (clockwise rotations about Y and Z, respectively, and performed in that order). The quantities $Z_\odot$, φ_{bar} and θ_{bar} are free parameters in Freudenreich's model, while $R_\odot$ is taken to be 8.5 kpc. Coordinate transformations from heliocentric spherical to Cartesian coordinates in the disk and bar frames are given in appendix A.

1.2 Sky Survey Data

Before the DIRBE maps can be used to derive the model, they must be processed to eliminate zodiacal light (sunlight scattered by ice and dust in the plane of the solar system) and point sources (most of which are bright stars close to the sun).

Freudenreich uses a detailed model of interplanetary dust to eliminate zodiacal light, leaving acceptably small (but not negligible) residuals. While the M band suffers most from zodiacal light, residuals were $\sim 0.01\,\mathrm{MJy\,sr^{-1}}$ for all maps (compared to the mean intensity $\sim 0.15\,\mathrm{MJy\,sr^{-1}}$ for $|b| > 20°$ in the L band). Freudenreich does not describe the zodiacal light model, but refers to a model proposed by another study [6] which used the same data from DIRBE.

Point sources pose a problem, as the model cannot hope to predict the location of randomly distributed local stars, but only the apparently continuous emission of the entire galactic structure. Point sources therefore need removal from the maps by use of a "de-starring" method, before they can be used to fit the model. This process is described in detail in Freudenreich (1996) [2, §2.3], and is not described in this report. The processed DIRBE images used in this report were prepared and kindly provided by Dylan Paterson (see acknowledgements). To show the effect of this process, the DIRBE L band before and after zodiacal light and point-source removal is shown in figure 1.1.

An implication of the point source removal is that local stars are effectively deleted, and their absence requires accounting for before the model is fitted. Freudenreich adds a parameter D_ν^{min} for this purpose, which, for a distance $s < D_\nu^{\mathrm{min}}$ along any line-of-sight in the model, the luminous density is set to zero. This accounts for the point source removal while assuming all removed sources exist within a 'deletion distance' D_ν^{min} from the sun [2, §2.3]. After parameters are fitted to the data, the 'deletion bubble' can be removed from

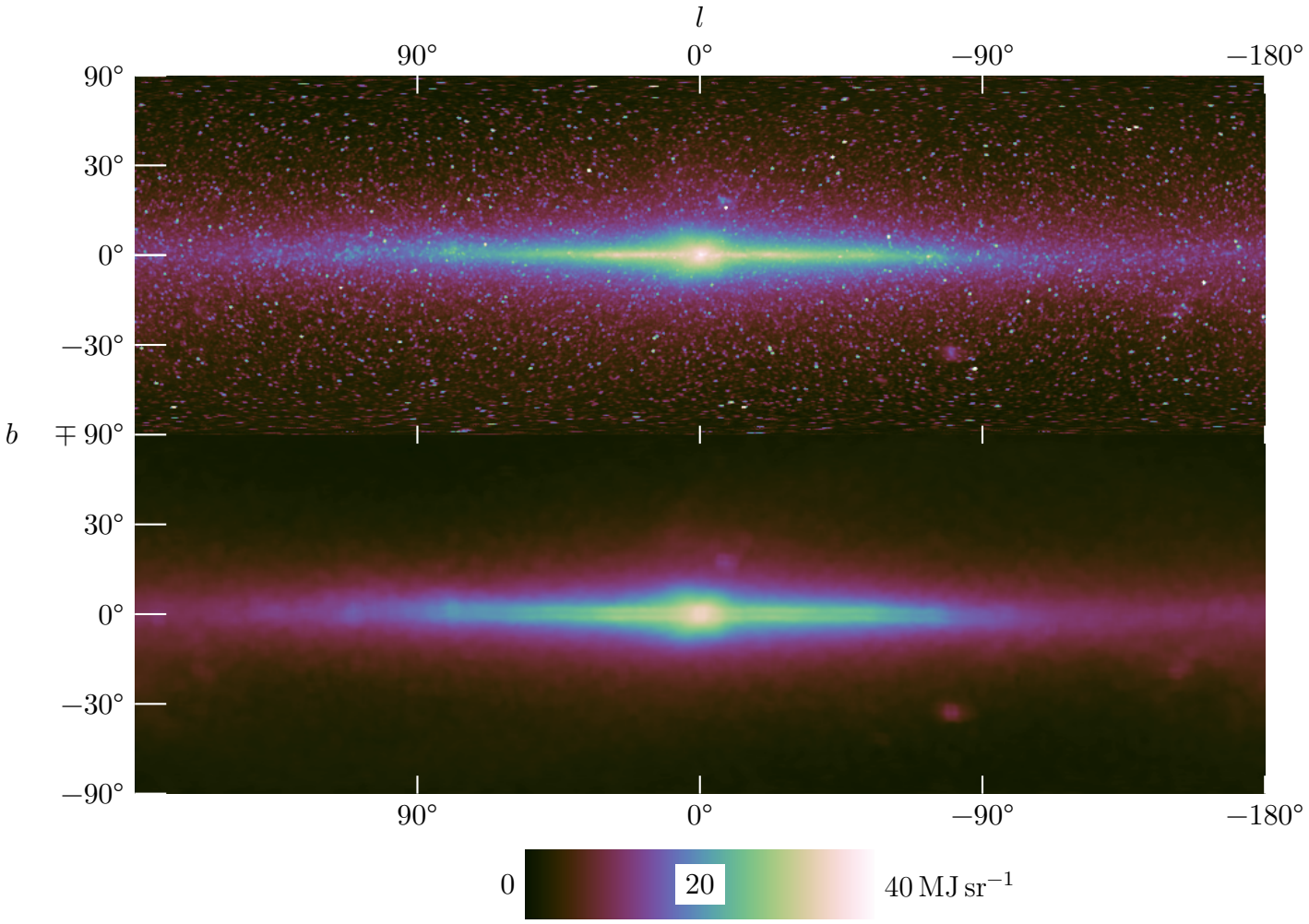

Figure 1.1: DIRBE maps of the L band (3.5 µm) before (*above*) and after (*below*) point source removal. The horizontal axis is longitude, l, and the vertical is latitude, b (under equal area projection).

the model, as it is not physically relevant.

1.3 Photometry

Freudenreich's model describes the overall spacial variance of the *spectral volume emissivity* of matter throughout the galaxy (at an infrared frequency ν). From the model of spectral volume emissivity can be acquired a model of the apparent spectral intensity as observed from Earth, for a given near-infrared band. The total emissivity is modelled as the net effect of the emissivities of the stellar disk, $\varepsilon_\nu^{\text{disk}}$; dust layer, $\varepsilon_\nu^{\text{dust}}$; and galactic bar, $\varepsilon_\nu^{\text{bar}}$. All three components embody illumination due to stars or heated interstellar gas, and the dust layer $\varepsilon_\nu^{\text{dust}}$ also acts to incorporate any illumination by diffuse scattering of light [3, §3.2]. (Freudenreich uses the symbol ρ to denote volume emissivity, but in the interest of clarity this report follows the conventions used throughout *Radiative Processes in Astrophysics*

[4], symbolising volume emissivity as ε and reserving ρ for density.)

The net volume emissivity ε_ν has units $\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$, and the observed intensity, I, has units $\mathrm{MJy\,sr^{-1}}$ of surface brightness,[2] so that the observed intensity is related to the volume emissivity in a relationship of the form

$$I_\nu(l,b) \;=\; \int \mathrm{d}s\; \varepsilon_\nu(l,b,s), \tag{1.1}$$

where the integration is performed over a line-of-sight from the observer to infinity, through the emissive volume.

However, in reality, extinction occurs as light is partially absorbed and scattered by dust and intragalactic matter. Freudenreich uses the dust layer to calculate extinction, encompassed in an absorption term, τ_ν. The extinction of light per unit distance travelled is proportional to the dust density ρ_ν^{dust} (not the dust emissivity, $\varepsilon_\nu^{\mathrm{dust}}$), at that point. Since occlusion is accumulative over distance, the extinction term for a point (l,b,s) as observed from the solar system is proportional to the total dust density along the line-of-sight from the observer to that point.

$$\tau_\nu(l,b,s_0) \;\propto \int_0^{s_0} \mathrm{d}s\; \rho_\nu^{\mathrm{dust}}(l,b,s) \tag{1.2}$$

The absorption term is incorporated into equation 1.1 as

$$I_\nu(l,b) \;=\; \int \mathrm{d}s\; \varepsilon_\nu(l,b,s)\, e^{-\tau_\nu(l,b,s)}.$$

Freudenreich also includes an additive offset term to absorb extragalactic background (and possibly residual zodiacal light), so that the final form of the intensity at frequency ν is

$$I_\nu(l,b) \;= \delta_\nu + \int_0^\infty \mathrm{d}s\; (\varepsilon_\nu^{\mathrm{disk}} + \varepsilon_\nu^{\mathrm{dust}} + \varepsilon_\nu^{\mathrm{bar}})\, e^{-\tau_\nu(s)}, \tag{1.3}$$

where δ_ν is a free parameter [3, §3.1]. (Because of the point source removal described in section 1.2, the lower limit of integration may be replaced with D_ν^{min}—another free parameter—when comparing with the processed sky maps.)

[2]The jansky, Jy, measures spectral flux density, or power per unit frequency bandwidth of light per unit area of aperture. Thus, the units $\mathrm{MJy\,sr^{-1}}$ are apt to measure the observed 'brightness' of a celestial point, and are the units of the DIRBE data sets.

2 Parametric Model

The Galactic Disk

Freudenreich's model of the galactic disk emissivity, $\varepsilon_\nu^{\text{disk}}$, is a circular cloud that is exponential in R and $\propto \text{sech}^2$ in Z, with a warping of the disk above and below the galactic plane, $Z = 0$. The warping is specified by the mean Z-component of the disk, $\bar{Z}$, modelled by a cubic polynomial in $u \equiv R - R_w$,

$$\bar{Z}(R,\theta) = (c_1 u + c_2 u^2 + c_3 u^3)\,\sin(\theta - \theta_w)$$

where free parameters are highlighted.[1] For $R < R_w$, the disk is made flat (i.e., $\bar{Z} = 0$), and for $R > R_w$, there is a nodal line along $\theta = \theta_w$. The disk is warped symmetrically in opposite directions on opposite sides. The disk is also given an elliptic hole, whose major axis is shared with the galactic bar. If the eccentricity of the hole is ε, and if $R_H^2 = X'^2 + (\varepsilon Y')^2$, then the hole is modelled by

$$H(R,\theta) = 1 - e^{-(R_H/O_R)^{O_N}}, \tag{2.1}$$

which approaches zero inside the hole and unity outside. Putting these together, using scale lengths h_r and h_z for the R and sech^2 dependencies, the full expression for the disk emissivity is

$$\varepsilon_\nu^{\text{disk}} = \varepsilon_{\nu,0}^{\text{disk}} H e^{-R/h_r}\,\text{sech}^2\left(\frac{Z - \bar{Z}}{h_z}\right), \tag{2.2}$$

where $\varepsilon_{\nu,0}^{\text{disk}}$ is also a free parameter[2], representing the maximum value of $\varepsilon_\nu^{\text{disk}}$ at the galactic centre, if there was no hole. The scale height parameter h_z is free for $R < R_{\max}$, but 0.5 kpc further out, in order to truncate the disk. [3, §3.1]

The Dust layer

The dust layer (which models the interstellar medium comprised of gas and molecular dust) is responsible for extinction (absorption and scattering) and emission of diffuse light. Absorption is proportional to the dust density, ρ_ν^{dust}, which is given the same form as the disk, but with its own parameters, so that

$$\rho_\nu^{\text{dust}} = \sigma_{\nu,0}^{\text{dust}} H^d e^{-R/h_r^d}\,\text{sech}\left(\frac{Z - x^d \bar{Z}}{h_z^d}\right), \quad [3, §3.2]. \tag{2.3}$$

[1] Hereinafter, free parameter symbols are highlighted on their first appearance.

[2] Freudenreich names these parameters with a (confusing) functional notation $\rho_\nu^{\text{disk}}(0)$ instead of $\varepsilon_{\nu,0}^{\text{disk}}$.

The warping of the dust layer is a scaled version of the disk warping, by a factor x^d. The hole, H^d, has the same form as equation 2.1, but with its own dust parameters O_R^d and O_N^d.

Freudenreich lets scattered light be accounted for in the emissivity of the dust layer, which is assumed to be proportional to the product of the dust density ρ_ν^{dust} and stellar emissivity $\varepsilon_\nu^{\text{disk}}$, so that

$$\varepsilon_\nu^{\text{dust}} = \varepsilon_{\nu,0}^{\text{dust}} H H^d e^{-R/h_r - 2R/3h_r^d} \operatorname{sech}^{10/3}\left(\frac{|Z - x^d \bar{Z}|}{h_z^d}\right), \quad [3, §3.2].$$

Dust absorption

Extinction is simulated in the way outlined in equations 1.2 and 1.3, with an appropriate constant of proportionality. Freudenreich expresses extinction in the J band as magnitudes of extinction per kiloparsec, A_J, and this quantity is allowed to vary as a free parameter. For longer wavelengths (K, L, and M), a power law in wavelength is used (instead of introducing superfluous free parameters for extinction at other wavelengths),

$$A_\lambda = A_J \left(\frac{\lambda}{\lambda_J}\right)^{-\alpha}, \tag{2.4}$$

where $\lambda_J = 1.25\,\mu\text{m}$, and α is free. Using this power law, the absorption term can be fully expressed as

$$\tau_\nu(l, b, s_0) = A_\lambda \cdot \frac{\ln(10)}{2.5} \cdot \int_0^{s_0} \mathrm{d}s \, \frac{\rho_\nu^{\text{dust}}(l, b, s)}{\rho_\nu^{\text{dust}}(0)}, \tag{2.5}$$

where $\rho_\nu^{\text{dust}}(0) = \rho_\nu^{\text{dust}}(0, 0, 0)$ is the dust density evaluated at the solar system, and the $\ln(10)/2.5$ coefficient accounts for the base-ten logarithmic magnitude scale[3] used by A_J, converting it to the natural base, e.

The Galactic Bar

The galactic bar is modelled as a 'generalised' ellipsoid, of the form $x^C + y^C + z^C = r^C$, where the *shape factor* C is not necessarily two. For a perfect ellipsoid, $C = 2$; for $C < 2$, the bar is more octahedral; and for $C > 2$, it is more cuboid.

Freudenreich lets the effective radius within the bar, R_s, be a generalised ellipse, with independent shape factors in the Z'-direction (face-on) and X', Y'-directions (edge-on) named $C_\perp$ and $C_\parallel$, respectively.

$$R_s^{C_\parallel} = \left[\left(\frac{|X'|}{a_x}\right)^{C_\perp} + \left(\frac{|Y'|}{a_y}\right)^{C_\perp}\right]^{C_\parallel/C_\perp} + \left(\frac{|Z'|}{a_z}\right)^{C_\parallel} \tag{2.6}$$

[3] This arises from the historical convention of apparent magnitude being defined as $m \equiv -2.5 \log_{10}(I/I_0)$.

The generalised radius, R_s, is used to model the radial dependence of the bar. If we call the radial dependence f, then the bar model is

$$\varepsilon_\nu^{\text{bar}} = \varepsilon_{\nu,0}^{\text{bar}}\, f(R_s) \qquad R \le R_{\text{end}} \tag{2.7}$$

$$\varepsilon_\nu^{\text{bar}} = \varepsilon_{\nu,0}^{\text{bar}}\, f(R_s)\, e^{-(R-R_{\text{end}})^2/{h_{\text{end}}}^2} \qquad R > R_{\text{end}} \tag{2.8}$$

where the model is multiplied by a Gaussian with scale length h_{end} for $R > R_{\text{end}}$, in order to truncate the bar.

Note that, in addition to the free parameters above, there are the pitch φ_{bar} and angle θ_{bar} of the bar, which define the $X'Y'Z'$-frame.

2.1 Bar Variations

It is in the choice of the radial dependence of the bar, f, that Freudenreich proposes three alternative functional forms. The model variations are named the S, E and P models, and their defining radial dependencies are

$$\begin{aligned} &\text{Model S} && f(R_s) = \operatorname{sech}^2 R_s\,, \\ &\text{Model E} && f(R_s) = \exp\left(-R_s^n\right), \end{aligned} \tag{2.9}$$

$$\text{Model P} \qquad f(R_s) = \left[1 + \left(R_s/R_c\right)^n\right]^{-1}, \tag{2.10}$$

where each model has its own independently fitted set of parameters.

Freudenreich typeset the model E function as $\exp(R_s^{-n})$ in [3, §3.3], however, such a function is unbounded and does not remotely resemble $\operatorname{sech}^2 R_s$, so it is easily identified as an error. In a more recent paper involving the shape of the galactic bar by Simion (2017) [7, §4.4], a model E function is used with the form $\exp(-R_s^n/2)$, which differs from equation 2.9 only by a constant which is absorbed by other free parameters. This implies Freudenreich's error was a typographical mutation of equation 2.9. Freudenreich also erroneously typeset the model P function as its own reciprocal; $1 + (R_s/R_c)^n$. This particular function is also unbounded and easily identified as a mistake. A comparison of each radial function is shown in figure 2.1. The P model appears to have a brighter and flatter bulge, while the model E has the most sharply defined centre.

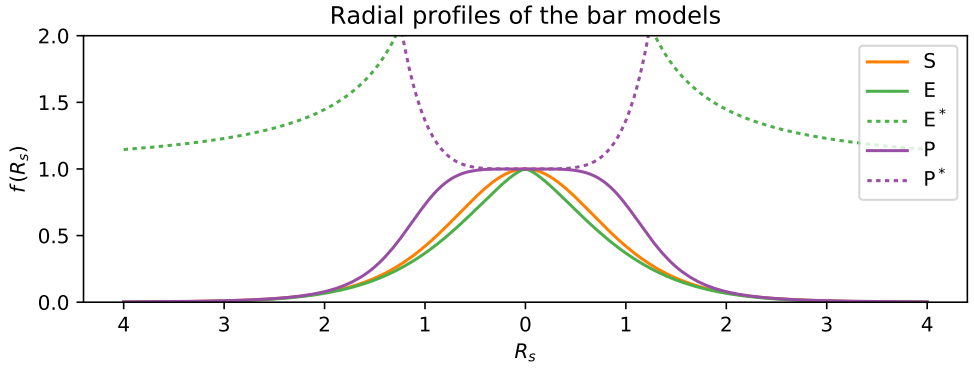

Figure 2.1: Plots of the radial dependencies of the bar for the S, E and P models. The E* and P* models (dotted) are as they are printed in Freudenreich (1998).

3 Visualising the model

3.1 Method of Computation

To produce an image of the model, the spectral intensity $I_\nu(l,b)$ of equation 1.3 must be evaluated at each pixel. Evaluating $I_\nu(l,b)$ can be expensive, because it is a line-of-sight integral whose integrand, $\varepsilon_\nu^\Sigma e^{\tau_\nu(s)}$, depends on the absorption term, $\tau_\nu(s)$, which is itself a line-of-sight integral. Since an image of the (l,b)-plane typically contains $886 \times 442 \approx 400\,000$ pixels, it is desirable to have an efficient computation method for each sky coordinate.

The method used in this report to compute the intensity at a single pixel at (l,b) consists of two steps. The first step is to determine the absorption term, $\tau_\nu(s)$, for each s in a linearly increasing sequence $\{0, s_1, s_2, ..., s_n\}$, where $s_k = k\,\mathrm{d}s$ for a small $\mathrm{d}s$. (The number of samples, n, is chosen so that s_n is beyond the outer limit of the model; $s_n \gtrsim 25\,\mathrm{kpc}$.) This can be done quickly by taking advantage of the accumulative nature of dust absorption; the absorption term at a certain distance s_i is the sum of the dust density at that point and the absorption term at the previous depth,

$$\tau_\nu(s_i) = \tau_\nu(s_{i-1}) + \rho_\nu^{\mathrm{dust}}(l,b,s_i)\,\mathrm{d}s,$$

in the limit $\mathrm{d}s \to 0$. Therefore, beginning at $\tau_\nu(0) = 0$, the absorption term, $\tau_\nu(s)$, can be computed for a set of discrete values along a single line-of-sight while only needing to evaluate the dust density model ρ_ν^{dust} once per sample, summing accumulatively.

The second step is to evaluate the total volume emissivity ε_ν^Σ at each point (l,b,s_i), for $i \in \{0, ..., n\}$, using the pre-computed absorption term at each i.

$$I_\nu(l,b) = \sum_{i=0}^{n} \varepsilon_\nu^\Sigma(l,b,s_i)\, e^{-\tau(s_i)}\,\mathrm{d}s$$

This results in the spectral intensity, $I_\nu(l,b)$, without ever having to evaluate a model function (ρ_ν^{dust}, $\varepsilon_\nu^{\mathrm{disk}}$, $\varepsilon_\nu^{\mathrm{dust}}$, or $\varepsilon_\nu^{\mathrm{bar}}$) at the same coordinates more than once. Applying this to each pixel in a (l,b)-grid results in an image of the model that can be compared to DIRBE maps.

This algorithm was implemented in multi-threaded C code and compiled with Freudenreich's fitted parameters, which are listed in table B.1. Most of these parameters are taken from table 3 of Freudenreich (1998) [3], and some (R_w and the warp coefficients) are listed in figure 12, while the deletion distances $D_\nu^{\min}$ and additive offsets δ_ν are given in [3, §5.1]. (Freudenreich (1998) erroneously typesets the dust scale height parameter h_ν^d in units of pc instead of kpc. Using parsecs produces a model that is clearly broken, while kiloparsecs works as expected.)

This method essentially applies Riemann summation to evaluate the two line-of-sight integrals, which is not a technique known for its accuracy. However, since the model does not contain finely detailed functions, the final intensity $I_\nu(l, b)$ converges as n increases sufficiently quickly. With $n \approx 500$, the produced image differs from a ground truth by no more than 0.01% at any pixel, and the program takes $\sim 8\,\mathrm{s}$ to execute on a laptop. A simple way to improve the efficiency of this method (which was not used in this report) would be to use a non-linear set of distances s_i. By decreasing $\mathrm{d}s$ where the model is more detailed, the quality of the Riemann sum in those regions is improved; and by increasing $\mathrm{d}s$ where the model is more flat, there is negligible increase in error. Letting $\mathrm{d}s$ vary with the gradient of the model in this way lets the total number of samples be reduced without loosing numerical precision, in turn improving efficiency.

3.2 Comparison & Residuals

Freudenreich's models can not be directly compared to the unprocessed DIRBE maps, because the presence of point sources in observed data unfairly disagrees with the continuous model. Rather, a comparison is drawn between the model and the de-starred DIRBE maps (described in section 1.2), after they have been offset to compensate for the deletion of nearby stars. Recall that the de-starring process was assumed to affect the DIRBE maps as if point sources within the deletion radius $D_\nu^{\min}$ were eliminated. By computing the fitted model only within the deletion radius, $0 \leq s < D_\nu^{\min}$, the offset required to correct the de-starred DIRBE maps is acquired. After adding this offset to the DIRBE maps, Freudenreich's models may be directly compared to them, and fair residuals may be computed.

The S, E and P models were computed in the L band by the method described in section 3.1, and their contours are shown in figure 3.1, superimposed over the (calibrated) DIRBE L band. In an effort to recreate figure 3 in Freudenreich (1998) (replicated here in figure 3.2), the same contour levels were used. (Freudenreich omitted the value of the fifth contour—$0.37\,\mathrm{MJy\,sr^{-1}}$—in the caption of figure 3 in [3], but its omission is easily noticed if the provided contour levels are plotted in log-space; they form a line with a definite gap where the fifth contour level was missed.)

While many features are shared between the two figures, the resemblance is not striking. Freudenreich's figure (3.2) displays a superior agreement between the model and data, while figure 3.1 exhibits larger disagreement, especially for the median contours, $10° < |b| < 30°$. The quality of fit along the galactic equator $b = 0°$ closely resembles Freudenreich, and the fit is strong around the bulge ($\leq 10°$ radius about the galactic centre). At low latitudes ($b \approx -60°$), there seems to be a systematic disagreement that is not found in Freudenreich, in which the data contours are too low for positive longitudes and too high for negative longitudes. This error, proportional to $\sin l$ may be due to residual zodiacal light.

However, if the DIRBE maps are ignored, the model S itself is identical to Freudenreich's, as can be seen by the model contours in figures 3.1–3.2. This implies that most of the discrepancy is due to differences in how the DIRBE data was processed, not how the model S was calculated. Freudenreich (1998) does not show the model E or P contours (since the model S is the preferred model [3, §5.4]), but they are included in figure 3.1. The S and E models are nearly identical, even in the bulge region where any differences in bulge shape should be apparent. The P model differs the most, and is the model with

the worst fit [3, §4]. The P model is most similar at low latitudes and high longitudes ($b \approx 0^\circ \ \wedge \ |l| < 45^\circ$), but is dimmer at high latitudes $|b| > 20^\circ$. Near the bulge, the P model is brighter for small longitudes $l \approx 0^\circ$ and dimmer for low latitudes $l \approx 0^\circ$.

Relative residuals were computed for the model S as $|\text{data} - \text{model}|/\text{data}$ with the calibrated DIRBE L band, and are shown in figure 3.3. Immediately noticeable is the proportionality of the error with latitude, $|b|$. There is also a significantly higher average error within the ecliptic band, outlined by the black contours at $|\beta| = \pm 15^\circ$, inferring that zodiacal light was not completely eliminated from the DIRBE L band. Regions of extremely large error ($> 40\%$) were identified[1] as major point sources which Freudenreich simply masked out during model fitting.

[1]Sources were identified with the help of the *Aladin Sky Atlas* developed at CDS, Strasbourg Observatory, France, which is freely available at `http://aladin.u-strasbg.fr`.

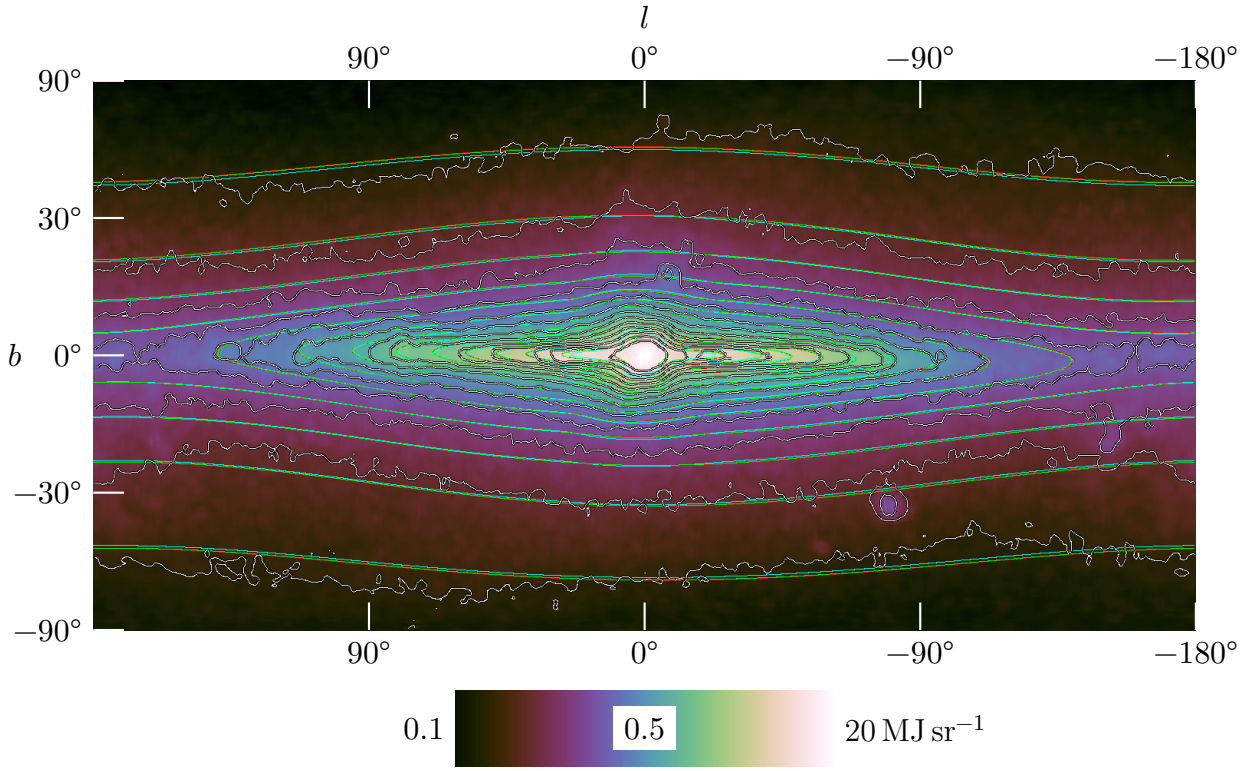

Figure 3.1: Contour map of the DIRBE L band after point source removal (white) with the contours of the S (green) and E (red) and P (cyan) models superimposed. Contour levels are 0.09, 0.13, 0.18, 0.26, 0.37, 0.52, 0.73, 1.04, 1.47, 2.08, 2.95, 4.17, 5.91 and 8.36 MJy sr^{-1}.

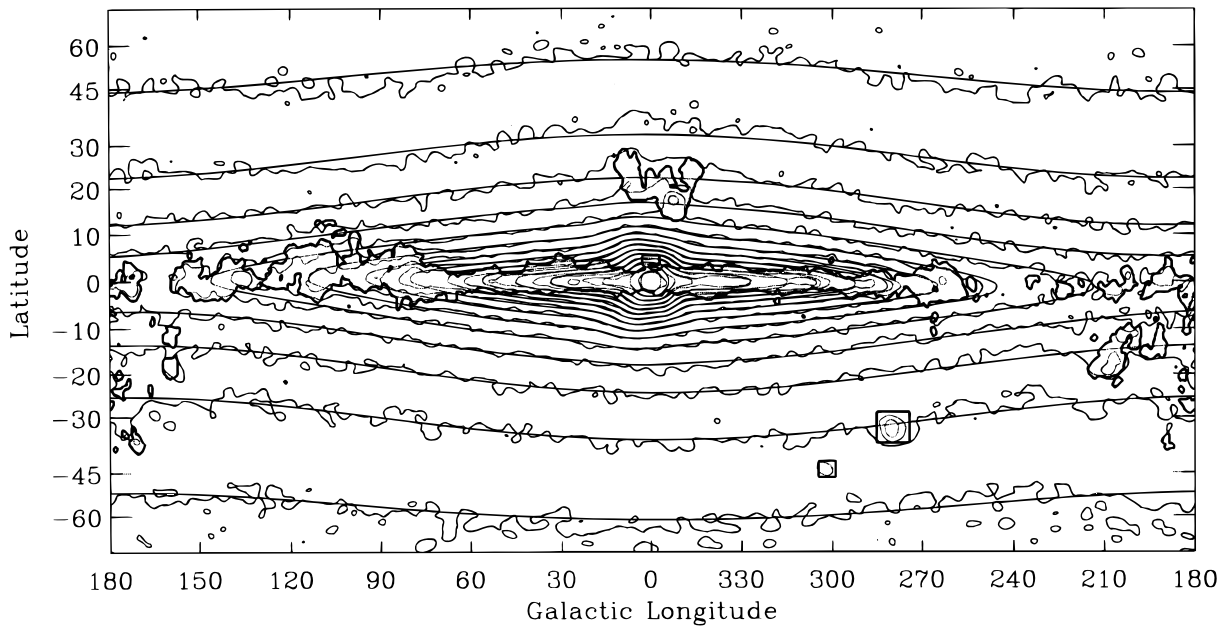

Figure 3.2: Freudenreich (1998) figure 3, showing the model S contours superimposed against processed data in the L band. Contour levels are as in figure 3.1. The thick contour shows the mask used while fitting the model.

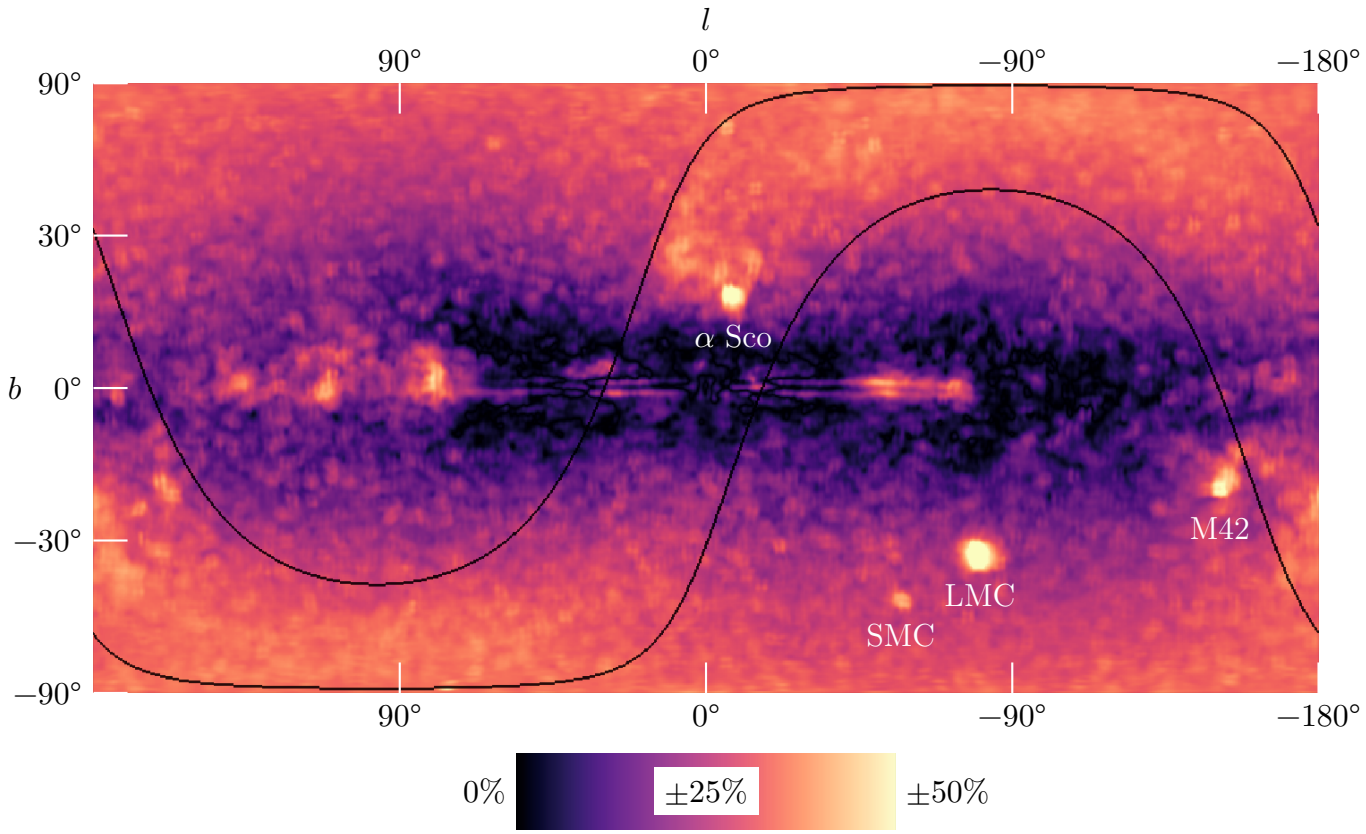

Figure 3.3: Model S residuals in the L band, showing the magnitude of the difference relative to the data. The three greatest residual spots are identified as bright sky objects; the Large and Small Magellanic Clouds (LMC & SMC), the Orion Nebula (M42) and the bright star Antares (α Sco).

4 Conclusions

Freudenreich's model was used to produce an intensity map in the L band that was able to be compared to processed DIRBE observations. While the S model was successfully replicated from Freudenreich (1998), the DIRBE L band was not processed an identical way. There was evidence of significant zodiacal light residue in the processed DIRBE L band, which caused disagreement between the model and the data.

The parametric model itself and the method of rendering it were explained in more detail than in Freudenreich (1998). The functional forms of the radial dependencies of the E and P models listed in Freudenreich (1998) were corrected with reference to Simion (2017) [7]. Other mistakes, such as the omitted contour in Freudenreich (1998) figure 3 and the incorrect unit for the dust scale height parameter, h_z^d, were corrected.

For these reasons, this report will be useful to anyone who is studying the galactic structure and intends to interpret Freudenreich's method, model and results.

Acknowledgements

I would like to thank Chris Gordon of the School of Physical and Chemical Sciences, University of Canterbury for supervising this project. I am also thankful to Dylan Paterson for providing me with the processed DIRBE data used in this investigation.

Bibliography

[1] J. N. Bahcall. Star counts and galactic structure. *ARA&A*, 24:577–611, 1986.

[2] H. T. Freudenreich. The Shape and Color the Galactic Disk. *Astrophysical Journal*, 468:663–678, 1996.

[3] H. T. Freudenreich. A COBE Model of the Galactic Bar and Disk. *Astrophysical Journal*, 492:495–510, 1998.

[4] Rybicki Lightman. *Radiative Processes in Astrophysics*. Wiley-Interscience, 1979.

[5] A. K. Pandey, S. Sharma, and K. Ogura. The population in the background of open clusters: tracer of the Norma-Cygnus arm. *MNRAS*, 373:255–262, 2006.

[6] William Reach, Bryan Franz, Thomas Kelsall, and Janet L. Weiland. DIRBE observations of the zodiacal light. 348, 01 1996.

[7] I. T. Simion, V. Belokurov, M. Irwin, S. E. Koposov, C. Gonzalez-Fernandez, A. C. Robin, J. Shen, and Z.-Y. Li. A parametric description of the 3D structure of the Galactic bar/bulge using the VVV survey. *MNRAS*, 471:4323–4344, 2017.

A Coordinate Transformations

If $Z_\odot$ is nonzero, which is suggested by observation, then the galactic plane defined by $b = 0°$ is not the same as the plane defined by $Z = 0$. Instead, the two planes intersect along the Y-axis at an angle $\varphi_\odot = \arctan(Z_\odot/R_\odot)$. This rotation, along with the translation between the heliocentric frame and galactic frame, define a coordinate transformation. The transformation from galactic (l, b, s) coordinates to galactocentric (X, Y, Z) coordinates is therefore an affine transformation which be decomposed into the following intermediates. After transforming from spherical to Cartesian heliocentric coordinates,

$$\begin{aligned} X_\odot &= s\cos(l)\cos(b) \\ Y_\odot &= s\sin(l)\cos(b) \\ Z_\odot &= s\sin(b) \end{aligned}$$

the frame origin is translated to the galactic centre,

$$\begin{aligned} X'_\odot &= X_\odot - D_\odot \\ Y'_\odot &= Y_\odot \\ Z'_\odot &= Z_\odot \end{aligned}$$

where $D_\odot$ is the distance between the sun and galactic centre. Finally, the coordinates undergo a clockwise rotation of $\varphi_\odot$ about the $+Y$-axis, aligning the XY-plane with the galactic disk.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \cos\varphi_\odot & 0 & \sin\varphi_\odot \\ 0 & 1 & 0 \\ -\sin\varphi_\odot & 0 & \cos\varphi_\odot \end{bmatrix} \begin{bmatrix} X'_\odot \\ Y'_\odot \\ Z'_\odot \end{bmatrix}$$

Furthermore, the bar frame coordinates $\vec{r}' = (X', Y', Z')$ are a rotation of the galactocentric coordinates $\vec{r} = (X, Y, Z)$. The transformation $\vec{r} \to \vec{r}'$ can be expressed as the composition of two orthogonal rotations, $\vec{r}' = \mathbf{\Theta\Phi}\,\vec{r}$. The first rotation, $\mathbf{\Phi}$, is an anticlockwise rotation of φ_{bar} about the $+Y$-axis, incorporating the pitch of the bar from the galactic plane. The second rotation, $\mathbf{\Theta}$, is an anticlockwise rotation of θ_{bar} about $+Z$, aligning the X'-axis with the major axis of the ellipsoidal bar.

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta_{\text{bar}} & -\sin\theta_{\text{bar}} & 0 \\ \sin\theta_{\text{bar}} & \cos\theta_{\text{bar}} & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{\Theta}} \underbrace{\begin{bmatrix} \cos\varphi_{\text{bar}} & 0 & -\sin\varphi_{\text{bar}} \\ 0 & 1 & 0 \\ \sin\varphi_{\text{bar}} & 0 & \cos\varphi_{\text{bar}} \end{bmatrix}}_{\mathbf{\Phi}} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

B Parameters

Parameter		Unit	Model S	Model E	Model P
Sun coordinates	$R_\odot$	kpc	8.5	=	=
‖	$Z_\odot$	pc	16.12	16.50	15.66
Deletion distances	$D_J^{\min}$	pc	470	=	=
‖	$D_K^{\min}$	pc	520	=	=
‖	$D_L^{\min}, D_M^{\min}$	pc	560	=	=
Additive offsets	δ_J	$\mathrm{kJy\,sr^{-1}}$	84	=	=
‖	δ_K	$\mathrm{kJy\,sr^{-1}}$	64	=	=
‖	δ_L	$\mathrm{kJy\,sr^{-1}}$	31	=	=
‖	δ_M	$\mathrm{kJy\,sr^{-1}}$	14	=	=
Disk parameters					
Disk scale length	h_r	kpc	2.6009	2.601	2.567
Disk scale height	h_z	kpc	0.3420	0.3466	0.3440
Disk radius	$R_{\max}$	kpc	12.35	12.45	12.52
Nodal line angle	θ_w	°	+0.40	+0.83	−0.07
Disk emissivities	$\rho_{J,0}^{\mathrm{disk}}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	8.115	8.141	8.725
‖	$\rho_{K,0}^{\mathrm{disk}}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	6.707	6.740	7.000
‖	$\rho_{L,0}^{\mathrm{disk}}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	3.539	3.478	3.637
‖	$\rho_{M,0}^{\mathrm{disk}}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	1.759	1.724	1.796
Disk hole radius	O_R	kpc	2.912	2.910	3.294
Disk hole shape	O_N	1	1.705	1.572	1.585
Hole eccentricity	ε	1	0.822	0.905	0.910
Warp inner radius	R_w	kpc	4.34	4.17	4.47
Warp coefficients	c_1	1×10^{-3}	+11.18	+12.09	+5.64
‖	c_2	$\mathrm{kpc^{-1}} \times 10^{-3}$	−1.92	−1.97	−2.27
‖	c_3	$\mathrm{kpc^{-2}} \times 10^{-3}$	+0.795	+0.679	+0.993
Bar parameters					
Bar tilt angle	θ_{bar}	°	+13.98	+9.48	+13.51
Bar pitch angle	φ_{bar}	°	−0.05	+0.07	+0.02
Bar sizes	a_x	kpc	1.686	1.878	1.806
‖	a_y	kpc	.6429	.6512	.6418
‖	a_z	kpc	.4420	.4302	.4313
Bar cutoff radius	R_{end}	kpc	3.139	3.542	2.725
Bar falloff scale length	h_{end}	kpc	0.469	0.545	0.875
Bar face-on shape	$C_\perp$	1	1.588	1.597	1.655

Bar edge-on shape	$C_{\parallel}$	1	3.466	3.418	2.976
Bar emissivities	$\rho^{\rm bar}_{J,0}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	10.42	10.36	11.77
"	$\rho^{\rm bar}_{K,0}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	8.769	8.707	9.400
"	$\rho^{\rm bar}_{L,0}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	4.545	4.465	4.848
"	$\rho^{\rm bar}_{M,0}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	2.241	2.180	2.387
Dust parameters					
Dust scale length	h^d_r	kpc	3.020	3.320	3.376
Dust scale height	h^d_z	kpc (not pc)	0.205	0.2019	0.182
Dust warp multiplier	x^d	1	1.811	1.765	1.749
Local extinction factor	A_J	$\mathrm{J\,mag\,kpc^{-1}}$	0.0446	0.0558	0.0618
Dust hole radius	O^d_R	kpc	2.684	2.222	2.051
Dust hole shape	O^d_N	1	2.182	2.116	2.466
Dust emissivities	$\rho^{\rm dust}_{J,0}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	4.681	4.699	4.795
"	$\rho^{\rm dust}_{K,0}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	1.146	1.229	1.145
"	$\rho^{\rm dust}_{L,0}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	2.196	2.320	2.455
"	$\rho^{\rm dust}_{M,0}$	$\mathrm{MJy\,sr^{-1}\,kpc^{-1}}$	3.185	3.139	3.559

Table B.1: Model parameters from Freudenreich (1998).

C Code

Code that was used to compute the model and produce figures 3.1 and 3.3 is included in this appendix. Code was compiled with `clang -fopenmp -l cfitsio` to enable multi-threading and link the open source `cfitsio` library (available from `https://heasarc.gsfc.nasa.gov/fitsio/`), which is used for writing `.fits` files.

```c
/* dirbe.c - main file
 * Renders intensity map of the galactic bar and disk from
 *   model and produces a FITS file with appropriate header.
 * Written for PHYS391 by Joseph Wilson, 2018.
 */

/* Model Selection */
#define MODEL 'P'
#define BAND LBAND

/* Image resolution */
#define LPORTION 1  // set both to 1 for full image
#define BPORTION 1

#define LSIZE (int)(886*LPORTION)
#define BSIZE (int)(442*BPORTION)
#define SSIZE 300

/* Includes */
#include "model.h"
#include "ndarray.h"
#include "savefits.h"

real l_lim[2] = {deg2rad(180*LPORTION), deg2rad(-180*LPORTION)};
real y_lim[2] = {-BPORTION, BPORTION};

int main(int argc, char const *argv[])
{
    if (argc < 2) {
        printf("usage: %s output.fits\n", argv[0]);
        exit(0);
    }

    init_model();   // compute derived parameters

    real L[LSIZE], B[BSIZE], S[SSIZE];
    real dl = linspace(l_lim[0], l_lim[1], LSIZE, L);
    real ds = linspace(0, 3*'D_sun, SSIZE, S);

    // create equal-area B axis, sin(b) = y (CEA projection)
    real y[BSIZE];
    real dy = linspace(y_lim[0], y_lim[1], BSIZE, y);
    for (unsigned int i = 0; i < BSIZE; ++i) B[i] = asin(y[i]);


    /* Create coordinate mesh */
    printf("computing coordinates...\n");

```

```
    struct ndarray coords;
    coords.ndim = 4;
    coords.dims = (unsigned int[]){
        NCOORDS, SSIZE, LSIZE, BSIZE};                      // note axis order
    ndarray_new(&coords);

    #pragma omp parallel for    // enable multithreading
    for (unsigned long p = 0; p < coords.size; p += NCOORDS) {
        unsigned int indices[4];    // declaration inside for loop means per thread
        ndarray_indices(&coords, p, indices);
        lbs_to_xyzXYZ(L[indices[2]], B[indices[3]], S[indices[1]],
            &coords.data[p]);
    }


    /* Compute absorption factor */
    printf("computing τ(s)...\n");

    real sun_coords[NCOORDS];
    lbs_to_xyzXYZ(0, 0, 0, /*out:*/sun_coords);
    real σ_dust_0 = σ_dust(sun_coords);

    real A_λ = 'A_J*pow(WAVELENGTH[BAND]/WAVELENGTH[JBAND], -'α);
    real coeff = A_λ/2.5*log(10)/σ_dust_0;

    struct ndarray τ;
    τ.ndim = 3;
    τ.dims = (unsigned int[]){SSIZE, LSIZE, BSIZE};
    ndarray_new(&τ);

    // sum σ_dust along each line of sight, creating τ mesh
    #pragma omp parallel for    // enable multithreading
    for (unsigned long p = 0; p < τ.size; p += SSIZE) {     // traverse volume in (s,l,b) order
        real σ = 0;                                         // (0,l,s) is beginning s-line
        for (unsigned int k = 0; k < SSIZE; ++k) {
            σ += coeff*σ_dust(&coords.data[(p + k)*NCOORDS])*ds;
            τ.data[p + k] = σ;
        }
    }

    /* Integrate model
     * I_ν(l, b) = δ_ν + ∫ ds (Σ ρ_ν) exp(-τ(s))
     */
    printf("integrating model...\n");

    struct ndarray I;
    I.ndim = 2;
    I.dims = (unsigned int[]){LSIZE, BSIZE};
    ndarray_new(&I);

    #pragma omp parallel for    // enable multithreading
    for (unsigned long p = 0; p < I.size; ++p) {            // traverse lb-plane
        I.data[p] = 'δ_ν[BAND];
        unsigned long p_3d = SSIZE*p;
        while (++p_3d%SSIZE > 0) {                          // traverse s-line
            real ρ = ρ_sum(&coords.data[NCOORDS*p_3d]);
            I.data[p] += ρ*exp(-τ.data[p_3d])*ds;
        }
    }

    /* Export FITS image */

    const char *filename = argv[1];
    save_image(filename, FLOAT_IMG, TDOUBLE, &I);

    float refx = (LSIZE + 1.0)/2;
    float refy = (BSIZE + 1.0)/2;
    float lonstep = rad2deg(l_lim[1] - l_lim[0])/LSIZE;
    float latstep = rad2deg(asin((y_lim[1] - y_lim[0])/BSIZE));

```

```
    add_card("CRPIX1",  TFLOAT,  &refx,         "ref pixel x");
    add_card("CRPIX2",  TFLOAT,  &refy,         "ref pixel y");
    add_card("CTYPE1",  TSTRING, &"GLON-CEA",   "galactic lon, l (equal area)");
    add_card("CTYPE2",  TSTRING, &"GLAT-CEA",   "galactic lan, b (equal area)");
    add_card("CUNIT1",  TSTRING, &"deg",        "unit of galactic lon");
    add_card("CUNIT2",  TSTRING, &"deg",        "unit of galactic lat");
    add_card("CDELT1",  TFLOAT,  &lonstep,      "lon step size");
    add_card("CDELT2",  TFLOAT,  &latstep,      "lat step size");
    add_card("CREATOR", TSTRING, &"Joseph Wilson @ UC, PHYS391 2017-2018.", "");

    char comment[80];
    sprintf(comment, "model: %c, band: %c, ssize: %d", MODEL, "JKLM"[BAND], SSIZE);
    add_comment(comment);

    close_image();
    free(coords.data);
    free(τ.data);
    free(I.data);

    printf("done.\n");
    return 0;
}
```

```
/* ndarray.h
 * A very simple implementation of n-dimensional arrays,
 *   which are stored as normal 1D arrays contained within
 *   a struct with information about dimensionality.
 * Written for PHYS391 by Joseph Wilson, 2018.
 */

#ifndef NDARRAY_H
#define NDARRAY_H

#include <stdlib.h>
#include <stdio.h>
#include <assert.h>

struct ndarray {
    real *data;             // 1D array of data
    unsigned int ndim;      // number of dimensions
    unsigned int *dims;     // size along each dimension
    unsigned long size;     // total number of array elements
};

void ndarray_new(struct ndarray *a)
{  /* Given an ndarray with initialised .ndim and .dim,
    * compute .size and allocate memory for .data.
    */
    a->size = 1;
    for (unsigned int i = 0; i < a->ndim; ++i) a->size *= a->dims[i];
    a->data = (real *)malloc(a->size*sizeof(real));
}
long ndarray_index(struct ndarray *a, unsigned int coords[])
{  /* Given indices of an element along each dimension of
    * an ndarray, return the real index of the element
    * as stored in the internal .data array.
    */
    unsigned int i = a->ndim - 1;
    unsigned long p = coords[i];
    while (i--) p = p*a->dims[i] + coords[i];
    return p;
}
void ndarray_indices(struct ndarray *a, unsigned long p,
    unsigned int coords[]) // output
{  /* Given the real index of an element in the .data array,
    * set the indices of the element along each dimension.
    */
    for (unsigned int i = 0; i < a->ndim; ++i) {
        coords[i] = p%a->dims[i];
```

```
        p /= a->dims[i];
    }
}


typedef unsigned int dim;
void matmul(dim n, dim l, dim m,
    real A[n][l], real B[l][m],
    real C[n][m]) // output
{  /* In-place matrix multiplication.
    * A(n×l) × B(l×m) → C(n×m)
    */
    for (int i = 0; i < n; ++i) {
        for (int j = 0; j < m; ++j) {
            C[i][j] = 0;
            for (int k = 0; k < l; ++k) {
                C[i][j] += A[i][k]*B[k][j];
            }
        }
    }
}

#endif /* NDARRAY_H */
```

```
/* savefits.h
 * Utilities for writing fits files from 2D arrays.
 * Written for PHYS391 by Joseph Wilson, 2018.
 */

#include <string.h>
#include <stdio.h>
#include <fitsio.h>

#ifndef SAVEFITS_H
#define SAVEFITS_H

void print_error(int status)
{
    fits_report_error(stderr, status);
    exit(status);
}

fitsfile *fptr;
int status = 0;

fitsfile *save_image(const char filename[], int bitpix, int datatype, struct ndarray *image)
{  /* Save a 2D array as a fits file.
    */


    /* Create new file */
    printf("exporting image to '%s'...\n", filename);

    remove(filename);
    if (fits_create_file(&fptr, filename, &status))
        print_error(status);

    /* Create image */
    // convert dims from int[] to long[].
    long dimsl[image->ndim];
    for (int i = 0; i < image->ndim; ++i) dimsl[i] = image->dims[i];
    if (fits_create_img(fptr, bitpix, image->ndim, dimsl, &status))
        print_error(status);

    /* Write array */
    if (fits_write_img(fptr, datatype, 1, image->size, image->data, &status))
        print_error(status);

    return fptr;
```

```
}

void add_card(char label[], int datatype, void *data, char comment[])
{
    if (fits_update_key(fptr, datatype, label, data, comment, &status))
        print_error(status);
}

void add_comment(char comment[])
{
    if (fits_write_comment(fptr, comment, &status))
        print_error(status);
}

void close_image()
{  /* Close previously created fits file.
    */
    if (fits_close_file(fptr, &status))
        print_error(status);
}

#endif /* SAVEFITS_H */
```

```
/* model.h
 * Header file that invokes the correct parameter set,
 *   define global constants and includes utilities.
 * Written for PHYS391 by Joseph Wilson, 2018.
 */

#include <math.h>

// type definitions for semantic purposes
typedef double real;
typedef real kpc;
typedef real rad;
typedef real mat3[3][3];

// constants
#define JBAND 0
#define KBAND 1
#define LBAND 2
#define MBAND 3

real WAVELENGTH[] = {        // μm
        1.25,         // J band
        2.2,          // K band
        3.5,          // L band
        4.9,        // M band
};

#define NCOORDS 6

// angle unit conversions
#define deg2rad(deg) ((deg)*M_PI/180)
#define rad2deg(rad) ((rad)/M_PI*180)

// include parameters for current model
#ifndef MODEL
        #error Model not specified
#elif MODEL == 'S'
    #include "params_S.h"
#elif MODEL == 'E'
    #include "params_E.h"
#elif MODEL == 'P'
    #include "params_P.h"
#else
    #error "Unknown model"
#endif

```

```
47  /* Global parameters */
48
49  // from Freudenreich 1998 §5.1
50
51  kpc 'D_min[] = {
52      .470, // J band
53      .520, // K band
54      .560, // L band
55      .560, // M band
56  };
57
58  real/*MJy/sr*/ 'δ_ν[] = {
59      .084, // J band
60      .064, // K band
61      .031, // L band
62      .014, // M band
63  };
64
65
66  // include headers that depend on parameters
67  #include "params_derived.h"
68  #include "model_functions.h"
69  #include "coords.h"
```

```
1   /* coords.h
2    * Subroutines for transforming between spherical galactic
3    *   coordinates and galactocentric Cartesian coordinates.
4    * Written for PHYS391 by Joseph Wilson, 2018.
5    */
6
7   #ifndef COORDS_H
8   #define COORDS_H
9
10  #include <stdio.h>
11  #include <math.h>
12
13  #include "ndarray.h"
14
15  void lbs_to_xyz(real l, real b, real s,
16      real *x, real *y, real *z) // output
17  {  /* In-place spherical to Cartesian coordinate transform.
18      */
19      real c = cos(b);
20      *x = s*c*cos(l);
21      *y = s*c*sin(l);
22      *z = s*sin(b);
23  }
24
25  void lbs_to_xyzXYZ(real l, real b, real s,
26      real xyzXYZ[]) // output
27  {  /* In-place conversion from spherical galactic
28        coordinates (l, b, s) into Cartesian coordinates
29        in the galactic disk frame (x, y, z) and
30        galactic bar frame (X, Y, Z).
31      */
32
33      real *x = &xyzXYZ[0], *y = &xyzXYZ[1], *z = &xyzXYZ[2],
34           *X = &xyzXYZ[3], *Y = &xyzXYZ[4], *Z = &xyzXYZ[5];
35
36      // Transform from spherical to Cartesian coordinates.
37      lbs_to_xyz(l, b, s, x, y, z);
38
39      // Translate heliocentric frame to galactocentre.
40      *x -= 'D_sun;
41
42      // Perform rotations into disk and bar frames.
43      matmul(3, 3, 3, 'SUN_to_DISK_rot, *x, *y, *z, x, y, z);
44      matmul(3, 3, 3, 'DISK_to_BAR_rot, *x, *y, *z, X, Y, Z);
45  }
```

```
46
47  void DISKxyz_to_BARxyz(real xyz[],
48      real XYZ[]) // output
49  {
50      matmul(3, 3, 3, 'DISK_to_BAR_rot,
51           xyz[0],  xyz[1],  xyz[2],
52          &XYZ[0], &XYZ[1], &XYZ[2]);
53  }
54
55  #endif /* COORDS_H */
```

```
1   /* params_derived.h
2    * Header file that contains derived parameters of the model,
3    *   or other constants that are determined at run-time.
4    * Written for PHYS391 by Joseph Wilson, 2018.
5    */
6
7   #include <math.h>
8   #include <string.h> // memcpy
9
10  #include "matrix.h"
11
12  kpc 'D_sun;
13  rad 'φ_sun;
14
15
16  // Rotation matrices
17  mat3 'θ_bar_rot;
18  mat3 'φ_bar_rot;
19  mat3 'SUN_to_DISK_rot;
20  mat3 'DISK_to_BAR_rot;
21
22  void init_model()
23  {  /* Compute dependent derived quantities
24      */
25      'D_sun = hypot('R_sun, 'Z_sun);
26      'φ_sun = atan2('Z_sun, 'R_sun);
27      // 'φ_sun = 0;
28
29      double c, s;
30
31      // Set sun-to-disk frame rotation matrix;
32      // a counter-clockwise rotation about y
33      c = cos('φ_sun), s = sin('φ_sun);
34      memcpy('SUN_to_DISK_rot, (mat3){
35          { c, 0, s},
36          { 0, 1, 0},
37          {-s, 0, c},
38      }, sizeof(mat3));
39
40
41      // Set bar angle rotation matrix;
42      // a clockwise rotation about z
43      c = cos('θ_bar), s = sin('θ_bar);
44      memcpy('θ_bar_rot, (mat3){
45          { c,-s, 0},
46          { s, c, 0},
47          { 0, 0, 1},
48      }, sizeof(mat3));
49
50      // Set bar pitch rotation matrix;
51      // a counter-clockwise rotation about y
52      c = cos('φ_bar), s = sin('φ_bar);
53      memcpy('φ_bar_rot, (mat3){
54          { c, 0,-s},
55          { 0, 1, 0},
56          { s, 0, c},
57      }, sizeof(mat3));
58
```

```
    // Compose bar rotations into single matrix
    matmul(3, 3, 3, 'θ_bar_rot, 'φ_bar_rot, 'DISK_to_BAR_rot);
}
```

```
/* model_functions.h
 * Header file containing the parametric model functions.
 * Written for PHYS391 by Joseph Wilson, 2018.
 */

#if MODEL == 'S'
    #define RADIAL pow(cosh(R_s), -2)
#elif MODEL == 'E'
    #define RADIAL exp(-pow(R_s, 'n))
#elif MODEL == 'P'
    #define RADIAL 1/(1 + pow(R_s/'R_c, 'n))
#endif

real ρ_bar(real coords[6])
{  /* Volume emissivity of the bar.
    */
    real X  = coords[0], Y  = coords[1];
    real X' = coords[3], Y' = coords[4], Z' = coords[5];
    real R = hypot(X, Y);

    // ellipsoid bar
    real R_s = pow(fabs(X')/'a_x, 'C_xy) + pow(fabs(Y')/'a_y, 'C_xy);
    R_s = pow(R_s, 'C_z/'C_xy) + pow(fabs(Z'/'a_z), 'C_z);
    R_s = pow(R_s, 1/'C_z);

    // radial dependence
    real radial = RADIAL;

    // radial falloff
    real falloff = (R <= 'R_end) ? 1 : exp(-pow((R - 'R_end)/'h_end, 2));

    return 'ρ0_bar[BAND]*radial*falloff;
}

real ρ_disk(real coords[6])
{  /* Volume emissivity of the disk.
    */
    real X  = coords[0], Y  = coords[1], Z = coords[2];
    real X' = coords[3], Y' = coords[4];
    real R = hypot(X, Y);
    real θ = atan2(Y, X);

    // disk hole
    real R_H = hypot(X', 'ε*Y');
    real hole = 1 - exp(-pow(R_H/'O_R, 'O_N));

    // radial falloff
    real h_r = (R <= 'R_max) ? 'h_r : 'h_r_max;
    real falloff = exp(-R/h_r);

    // disk warping
    real u = R - 'R_w;
    real Z_bar = (R <= 'R_w) ? 0 : (('c[0] + ('c[1] + 'c[2]*u)*u)*u)*sin(θ - 'θ_w);
    real warp = pow(cosh((Z - Z_bar)/'h_z), -2);

    return 'ρ0_disk[BAND]*hole*falloff*warp;
}


real σ_dust(real coords[6])
{  /* Absorption of dust layer.
    */
    real X  = coords[0], Y  = coords[1], Z = coords[2];
    real X' = coords[3], Y' = coords[4];
    real R = hypot(X, Y);
```

```
    real θ = atan2(Y, X);

    // disk hole
    real R_H = hypot(X', 'ε*Y');
    real hole = 1 - exp(-pow(R_H/'Od_R, 'Od_N));

    // radial falloff
    real h_r = (R <= 'R_max) ? 'hd_r : 'h_r_max;
    real falloff = exp(-R/h_r);

    // disk warping
    real u = R - 'R_w;
    real Z_bar = (R <= 'R_w) ? 0 : (('c[0] + ('c[1] + 'c[2]*u)*u)*u)*sin(θ - 'θ_w);
    real warp = pow(cosh((Z - 'xd*Z_bar)/'hd_z), -2);

    return 'ρ0_dust[BAND]*hole*falloff*warp;
}

real ρ_dust(real coords[6])
{  /* Volume emissivity of dust layer.
    */
    real X  = coords[0], Y  = coords[1], Z = coords[2];
    real X' = coords[3], Y' = coords[4];
    real R = hypot(X, Y);
    real θ = atan2(Y, X);

    // disk and dust holes
    real R_H = hypot(X', 'ε*Y');
    real disk_hole = 1 - exp(-pow(R_H/'O_R, 'O_N));
    real dust_hole = 1 - exp(-pow(R_H/'Od_R, 'Od_N));

    // disk and dust falloff
    real falloff = exp(-R/'h_r - 2*R/(3*'hd_r));

    // disk warping
    real u = R - 'R_w;
    real Z_bar = (R <= 'R_w) ? 0 : (('c[0] + ('c[1] + 'c[2]*u)*u)*u)*sin(θ - 'θ_w);
    real warp = pow(cosh((Z - 'xd*Z_bar)/'hd_z), -10.0/3.0);

    return 'ρ0_dust[BAND]*dust_hole*disk_hole*falloff*warp;
}
```

```
/* params_S.h
 * Header file containing the fitted parameter values from
 *   Freudenreich (1998), table 3, model S (minimal mask).
 */

// Position of sun relative to galactic centre.
kpc 'R_sun = 8.5;
kpc 'Z_sun = 16.12e-3;

/* CHECK THIS: θ = ±13.79°? see params */
rad 'θ_bar = deg2rad(+13.98);           // bar tilt angle

/*
 * DISK PARAMETERS
 */

kpc 'h_r = 2.6009;                      // disk scale length
kpc 'h_z = 0.3420;                      // disk scale height

kpc 'R_max = 12.35;                     // disk radius
kpc 'h_r_max = 0.5;                     // disk scale truncation value

rad 'θ_w = deg2rad(0.40);               // warp line of nodes

real 'ρ0_disk[] = {                     // central disk emissivities
    8.115, // J band
    6.707, // K band
```

```
    3.539, // L band
    1.759, // M band
};

kpc 'O_R = 2.912;                       // disk-hole radius
real 'O_N = 1.705;                      // disk-hole power

real 'ε = 0.822;                        // hole axis ratio

// From Freudenreich 1998 fig. 12, (S).
kpc 'R_w = 4.34;                        // warp inner radius
real 'c[] = {                           // warp coefficients
    +0.01118,
    -0.00192,
    +.000795,
};

/*
 * BAR PARAMETERS
 */

rad 'φ_bar = deg2rad(-0.05);            // bar pitch angle

kpc 'a_x = 1.686;                       // bar size
kpc 'a_y = .6429;
kpc 'a_z = .4420;

kpc 'R_end = 3.139;                     // bar cutoff radius
kpc 'h_end = 0.469;                     // bar cutoff scale length

real 'C_xy = 1.588;                     // bar shape
real 'C_z = 3.466;


real 'ρ0_bar[] = {                      // central bar emissivities
    10.42,  // J band
    8.769,  // K band
    4.545,  // L band
    2.241,  // M band
};

//
//

/*
 * DUST PARAMETERS
 */

kpc 'hd_r = 3.020;                      // dust scale length

kpc 'hd_z = 0.205;                      // dust scale height

// 0.205 pc = 0.205e-3

real 'xd = 1.811;                       // dust warp factor

real 'A_J = 0.0898;                     // local extinction factor
real 'α = 1.987;                        // extinction index

kpc 'Od_R = 2.684;                      // dust hole radius
real 'Od_N = 2.182;                     // dust hole power

real 'ρ0_dust[] = {                     // dust central emissivities
    4.681, // J band
    1.146, // K band
    2.196, // L band
    3.185, // M band
};
```