SUPPLEMENTARY MATERIAL: A GENERALISATION OF IDEAL GAS MODELS FOR CAMERA TRAPS AND ACOUSTIC SENSORS

Contents

1. Supplementary Methods

1.1. **Introduction.** This supplementary methods derives all the models used in the paper. For continuity, the gas model derivation is included here as well as in the main text. The derivation of all models is included in the Python script S2.

1.2. **Gas model.** We assume that animals are in an homogeneous environment, and move in straight lines of random direction with velocity v. We allow that our sensor can detect animals at a distance r and that if an animal moves within this detection region they are detected with a probability of 1, independent of distance from the sensor while animals outside the region are never detected.

We then consider movement from the reference frame of the animals so that now, all animals are stationary and randomly distributed in space, while the sensor moves with velocity v. If we calculate the area covered by the sensor during the study period we can estimate the number of animals it should encounter. We calculate this as the average width of the sensor region p multiplied by v. The average width of the profile is the integral of the profile width over a full circle, divided by 2π. We use x_i to denote the focal angle which is the angle we integrate over. The subscript i distinguishes different angles (see Figure S1) but here we use x_1. As all models are bilaterally symetric, we can integrate over a half circle, and divide by π.

$$p\text{Gas} = \frac{1}{\pi}\int_{\frac{\pi}{2}}^{\frac{3\pi}{2}} 2r \, \mathrm{d}x_1 \qquad \text{eqn S1}$$

$$p\text{Gas} = 2r \qquad \text{eqn S2}$$

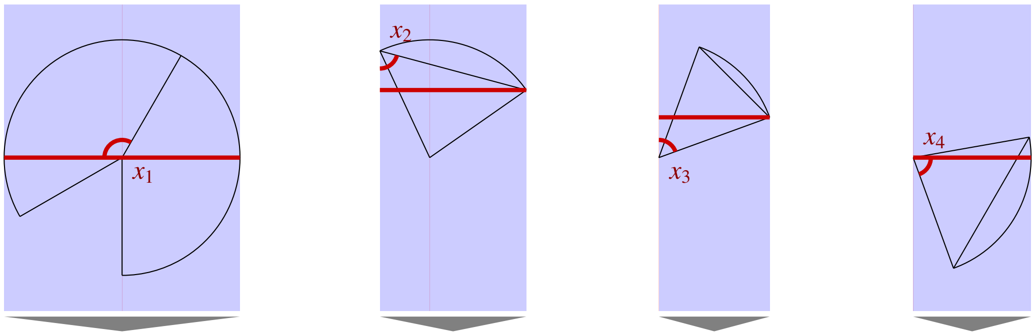
The number of expected encounters, z, for a survey of duration t, with an animal density of D is then

$$z = 2rvtD. \qquad \text{eqn S3}$$

However, in practice we have the opposite situation. We know the number of encounters and want to estimate the density. We do this be simply rearranging to get

$$D = z/(2rvt). \qquad \text{eqn S4}$$

For different values of θ and α, the only thing that changes is that the area covered per unit time is no longer given by $2rv$. Instead of the sensor having a diameter of $2r$, the sensor has a complex



(A) x_1 (B) x_2 (C) x_3 (D) x_4

FIGURE S1. The location of the focal angles $x_{i\in[1,4]}$. In these figures, the segment shaped detection region is shown in black. The width of this region is shown with a thick red line and a blue rectangle. The direction of animal movement is always downwards, as indicated by the grey arrow.

diameter that changes with approach angle. The rest of the derivation is just calculating this value for all values of α and θ. However, different regions of this two dimensional parameter space have noncontinuously different models, with different derivations. Therefore we have to identify the regions for which the derivation is the same, and then separately derive p for each region.

1.3. **Model p311.** p311 is very similar to the gas model except that as $\alpha \leq \pi$ the profile width is no longer $2r$ but is instead limited by the width of the animal call. We therefore get a profile width of $2r\sin(\alpha/2)$ instead (see Fig S2b).

$$p311 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}}^{\frac{3\pi}{2}} 2r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_1\right) \qquad \text{eqn S5}$$

$$p311 = 2r\sin\left(\frac{\alpha}{2}\right) \qquad \text{eqn S6}$$

1.4. **Model p22.** For regions with profiles that are more complex than a circle we need to explicitly write functions for the width of the profile for every approach angle. We then use these functions to find the average profile width for all approach angles by integrating across all 2π angles of approach and dividing by 2π.

There are three regions within cell p22. Note that p221 covers the area $\alpha = 2\pi$ as well as the triangle below it as these two models are specified exactly the same, rather than happening to have equal results.

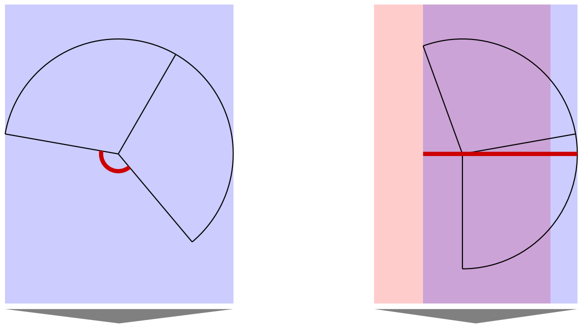
These models have up to five regions. 1) The profile width starts, from $x_1 = \frac{\pi}{2}$ as $2r$. 2) At $x_1 = \theta/2$, the right hand side of the profile cannot be r wide as the corner of the 'blind spot' (see Fig. S2a) limits its size to being $r\cos(x_1 - \theta/2)$ wide (see Fig. S3a).

3) The third profile is only found in p223. If $\alpha < 4\pi - 2\theta$, then at $x_1 = \theta/2$, when the profile is perpendicular to the edge of the blind spot, the whole right side of the profile is invisible to the sensor (see Fig. S3b). This gives a profile size of just r.

4) At some point, the sensor can detect animals once they have passed the blind spot giving a profile width of $r + r\cos(x_1 + \theta/2)$. From $x_1 = \pi$, if the animal call is wide enough to be detected in this area, this is the wider profile. This then defines the split between p221 and p222. In p221, with $\alpha > 3\pi - \theta$, the animal call is wide enough that at $x_1 = \pi$ the animal can already be detected past the blind spot and so this profile is used. In p222, with $\alpha < 3\pi - \theta$, the latter profile is reached at $5\pi/2 - \theta/2 - \alpha/2$ and is therefore dependant on the sizes of α and θ.

5) Finally, common to all three models, at $x_1 = 2\pi - \theta/2$ the profile becomes a full $2r$ once again.

1.4.1. *Model p221.* Model p221 exists within the area bounded by $\alpha \leq 2\pi$, $\theta \leq 2\pi$ and $\alpha \geq 3\pi - \theta$. It has four regions; it does not include the r profile at $x_1 = \pi$. Furthermore, θ is wide enough that the $r + r\cos(x_1 + \theta/2)$ profile starts at π. This then gives us



(A) Blind spot (B) Directional Call

FIGURE S2. A) Shows the area referred to as the 'blind spot'. B) For directional calls, with $\alpha < \pi$, the width of the profile can be limited by the call angle or by the detector region. The detector width is shown in blue, while the call width is shown as a red rectangle. Only where the two overlap, giving a purple area, can an animal be detected. Here we would say the right side of the profile is limited by the sensor, while the left side of the profile is limited by the call angle. The terms in equations would reflect this by containing α if call limited and containing θ if detector limited.

$$p221 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}}^{\frac{\theta}{2}} 2r\,\mathrm{d}x_1 + \int_{\frac{\theta}{2}}^{\pi} r + r\cos\left(-x_1 + \frac{\theta}{2}\right)\mathrm{d}x_1 + \int_{\pi}^{2\pi-\frac{\theta}{2}} r + r\cos\left(x_1 + \frac{\theta}{2}\right)\mathrm{d}x_1 + \int_{2\pi-\frac{\theta}{2}}^{\frac{3\pi}{2}} 2r\,\mathrm{d}x_1 \right) \quad \text{eqn S7}$$

$$p221 = \frac{r}{\pi}\left(\theta + 2\sin\left(\frac{\theta}{2}\right)\right) \quad \text{eqn S8}$$

1.4.2. *Model p222.* Model p222 is bounded by $\alpha \leq 3\pi - \theta$, $\alpha \geq 4\pi - 2\theta$ and $\alpha \geq \pi$. It is the same as p221 except that the third profile starts at $5\pi/2 - \theta/2 - \alpha/2$ instead of at π which is reflected in the different bounds in the second and third integral.

$$p222 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}}^{\frac{\theta}{2}} 2r\,\mathrm{d}x_1 + \int_{\frac{\theta}{2}}^{\frac{5\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}} r + r\cos\left(-x_1 + \frac{\theta}{2}\right)\mathrm{d}x_1 + \int_{\frac{5\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}}^{2\pi-\frac{\theta}{2}} r + r\cos\left(x_1 + \frac{\theta}{2}\right)\mathrm{d}x_1 + \int_{2\pi-\frac{\theta}{2}}^{\frac{3\pi}{2}} 2r\,\mathrm{d}x_1 \right) \quad \text{eqn S9}$$

$$p222 = \frac{r}{\pi}\left(\theta - \cos\left(\frac{\alpha}{2}\right) + \cos\left(\frac{\alpha}{2} + \theta\right)\right) \quad \text{eqn S10}$$

1.4.3. *Model p223.* Model p223 is bound by $\alpha \leq 4\pi - 2\theta$, $\alpha \geq \pi$ and $\theta \geq \pi$. It is the same as p222 except that it contains the extra profile with width r (third integral).

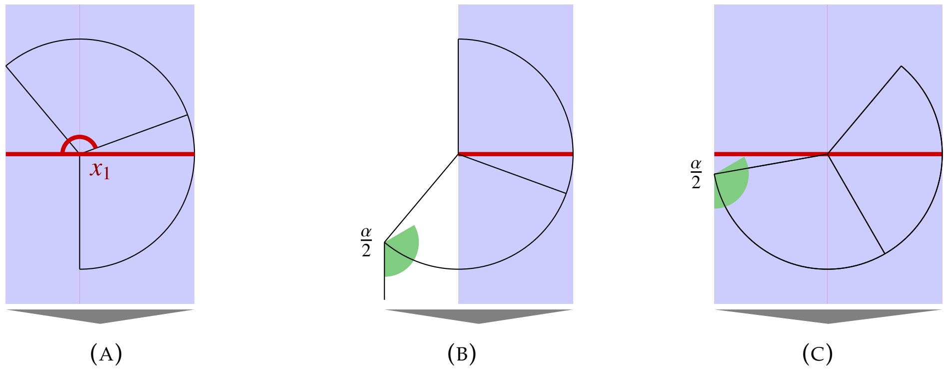


FIGURE S3. A) The second integral in p22 with width $r + r\cos(x_1 - \theta/2)$ B) The third integral in p223. The angle shown in red is $\alpha/2$. As it is small, animals to the right of the detector cannot be detected. C) After further rotation, $\alpha/2$ is now bigger than the angle shown and animals to the right of the detector can again be sensed.

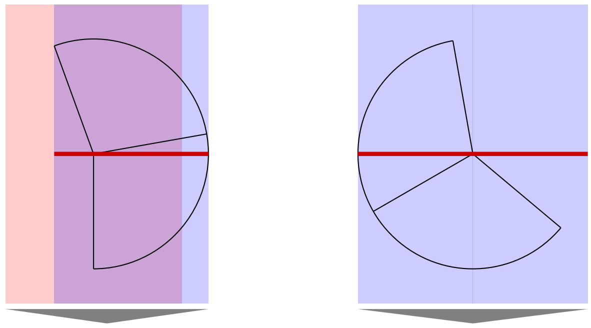
$$p223 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}}^{\frac{\theta}{2}} 2r \, dx_1 + \int_{\frac{\theta}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}} r + r\cos\left(-x_1 + \frac{\theta}{2}\right) dx_1 \right.$$
$$\left. + \int_{\frac{\pi}{2}+\frac{\theta}{2}}^{\frac{5\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}} r \, dx_1 + \int_{\frac{5\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}}^{2\pi-\frac{\theta}{2}} r + r\cos\left(x_1 + \frac{\theta}{2}\right) dx_1 + \int_{2\pi-\frac{\theta}{2}}^{\frac{3\pi}{2}} 2r \, dx_1 \right) \qquad \text{eqn S11}$$
$$p223 = \frac{r}{\pi}\left(\theta - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S12}$$

1.5. **Model p32.** Cell p32 contains three regions that differ in ways reminiscent of the models in p22. There are four possible profile widths. 1) As α is less than π the profile is smaller than $2r$, even when the sensor width is a full diameter. When this is the case, the profile width is instead $2r\sin(\alpha/2$. 2) Similar to p22, at a certain point the blind spot of the sensor area limits the profile width (see Fig. S4a). This gives a profile width of $r\sin(\alpha/2) + r\cos(x_1 - \theta/2)$. 3) Also similar to p22, there can be a point where the right side of the profile is 0 giving a profile width of $r\sin(\alpha/2)$. 4) If $\alpha \le 2\pi - \theta$, then at $\theta/2 + \pi/2 + \alpha/2$ the profile width become 0 (see Fig. S4b). This inequality distinguishes between p322 and p323. The profile $r\sin(\alpha/2)$ starts at $\theta/2 + \pi/2$ while at $5\pi/2 - \alpha/2 - \theta/2$ the profile returns to size $2r\sin(\alpha/2)$. If $\theta/2 + \pi/2 \ge 5\pi/2 - \alpha/2 - \theta/2$ we go straight into the $2r\sin(\alpha/2)$ profile and miss the $r\sin(\alpha/2)$ profile. p321 and p322 are seperated by this inequality which simplifies to $\alpha \le 4\pi - 2\theta$.

1.5.1. *Model p321.* p321 is bounded by $\alpha \ge 4\pi - 2\theta$, $\alpha \le \pi$ and $\theta \le 2\pi$. As $\alpha \ge 4\pi - 2\theta$, there is no $r\sin(\alpha/2)$ profile. As $\alpha \le 4\pi - 2\theta$, the profile returns to $2r\sin(\alpha/2)$ rather than going to 0.

$$p321 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}} 2r\sin\left(\frac{\alpha}{2}\right) dx_1 + \int_{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{5\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}} r\cos\left(-x_1 + \frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right) dx_1 + \int_{\frac{5\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{3\pi}{2}} 2r\sin\left(\frac{\alpha}{2}\right) dx_1 \right) \qquad \text{eqn S13}$$
$$p321 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + \cos\left(\frac{\alpha}{2} + \theta\right)\right) \qquad \text{eqn S14}$$

1.5.2. *Model p322.* p322 is bounded by $4\pi - 2\theta \le \alpha \le 4\pi - 2\theta$ and $\alpha \le \pi$. Therefore there is a $r\sin(\alpha/2)$ profile but no $0r$ profile.



(A) (B)

FIGURE S4. A) The third integral in p32. The right side of the profile is limited by the size of the sensor region (blue region) while the left side of the profile is limited by the size of the call angle (red region). The profile width is the purple region where these two overlap. B)

$$p322 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}} 2r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_1 + \int_{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}} r\cos\left(-x_1+\frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_1 \right.$$
$$\left. + \int_{\frac{\pi}{2}+\frac{\theta}{2}}^{\frac{5\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_1 + \int_{\frac{5\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{3\pi}{2}} 2r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_1 \right) \qquad \text{eqn S15}$$

$$p322 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S16}$$

1.5.3. *Model p323.* Finally p323 is bounded by $\alpha \leq 4\pi - 2\theta$, $\alpha \leq \pi$ and $\theta \leq \pi$. It is the same as p322 except that the profile becomes $2r$ rather than returning to $2r\sin(\alpha/2)$.

$$p323 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}} 2r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_1 + \int_{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}} r\cos\left(-x_1+\frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_1 + \int_{\frac{\pi}{2}+\frac{\theta}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_1 \right) \qquad \text{eqn S17}$$

$$p323 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S18}$$

1.6. **Model p131.** p131 is the first model with $\theta < \pi$. Whereas previously the focal angle has always been x_1, we now use different focal angles. x_2 and x_3 correspond to γ_1 and γ_2 in Rowcliffe *et al.* (2008) while x_4 is new. They are described in Fig. S1.

There are five different profiles in p131. 1) x_2 has an interval of $[\pi/2, \theta/2]$ which is from the angle of approach being directly towards the sensor until the profile is parellel to the left hand radius of the sensor segment. During this region the profile width is $2r\sin(\theta/2)\sin(x_2)$ which is calculated using the equation for the length of a chord (see Fig. S1b). Note that while rotating anti-clockwise (as usual) x_2 decreases in size. 2) From here, we examine focal angle x_4 (note that x_3 is used in later models, but is not relevant here.) The left side of the profile is a full radius while the right side is limited to $-r\cos(x_4-\theta)$ (see Fig. S5a). 3) At $x_4 = \theta - \pi/2$, the profile is perpendicular to the edge of the sensor area. Here, the right side of the profile is $0r$. 4) When $x_4 = \pi/2$ the angle of approach is from behind the sensor, but we can once again be detected on the right side of the sensor (see Fig. S5b). Therefore the width of the profile is $r - r\cos(x_4)$. 5) Finally, we enter the x_2 region, but from behind.

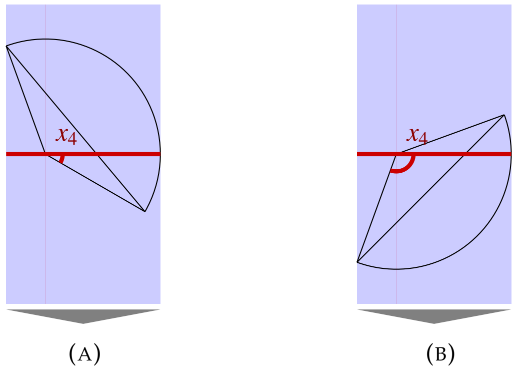


FIGURE S5. A) and B) The third and fourth profiles of p131. The left side of of both profiles is of width r while the right side is $-r\cos(x_4 - \theta)$ and $-r\cos(x_4)$ respectively.

$$p131 = \frac{1}{\pi}\left(\int_{\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 + \int_{0}^{-\frac{\pi}{2}+\theta} r - r\cos(-x_4+\theta)\,\mathrm{d}x_4 \right.$$
$$\left. + \int_{-\frac{\pi}{2}+\theta}^{\frac{\pi}{2}} r\,\mathrm{d}x_4 + \int_{\frac{\pi}{2}}^{\theta} r - r\cos(x_4)\,\mathrm{d}x_4 + \int_{\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 \right) \qquad \text{eqn S19}$$

$$p131 = \frac{r}{\pi}(\theta+2) \qquad \text{eqn S20}$$

1.7. **Model p23.** The models in cell p23 have the five potential profiles in p131 but not all profiles occur in each model, and the angle at which transitions occur are different. Furthermore, there is one extra profile possible. When approaching the sensor from behind, there is a period where the profile is r wide as in p131. At some point the right side of the profile becomes viable again. If this occurs in the x_4 region, the profile width becomes $r - r\cos(x_4)$ as in p131. However, as α is now less than 2π, the right side of the profile might not be viable until we are in the second x_2 region. In this case, when we first enter the second x_2 region, the profile has a width of $r\cos(x_2 - \theta/2)$. This occurs only if $\alpha \leq 3\pi - 2\theta$. This is inequality is found by noting that the right side of the profile become viable at $x_4 = 3\pi/2 - \alpha$ but the x_2 region starts at $x_4 = \theta$. The new profile in x_2 will only occur if $\theta < 3\pi/2 - \alpha/2$ which is rearranged to find the inequality above. This defines the boundary between p231 and p232.

As $\alpha \leq 2\pi$ it is possible that when the angle of approach is from directly behind the sensor the animal will not be detected at all. This is the case if $\alpha/2 \leq \pi - \theta/2$ as shown in Fig. S6a. This inequality defines the boundary between p232 and p233.

1.7.1. *Model p231.* p231 is bounded by $\alpha \geq 3\pi - 2\theta$, $\alpha \leq 2\pi$ and $\theta \leq \pi$.

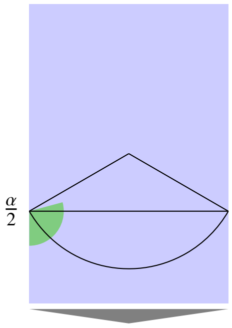
p231 has all five profiles as found in p131. However, the change from the r profile (third integral) to the $r - r\cos(x_4)$ profile (fourth integral) occurs at $x_4 = 3\pi/2 - \alpha/2$ instead of at $x_4 = \theta$.

$$p231 = \frac{1}{\pi}\left(\int_{\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 + \int_{0}^{-\frac{\pi}{2}+\theta} r - r\cos(-x_4+\theta)\,\mathrm{d}x_4 \right.$$
$$\left. + \int_{-\frac{\pi}{2}+\theta}^{\frac{3\pi}{2}-\frac{\alpha}{2}} r\,\mathrm{d}x_4 + \int_{\frac{3\pi}{2}-\frac{\alpha}{2}}^{\theta} r - r\cos(x_4)\,\mathrm{d}x_4 + \int_{\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 \right) \qquad \text{eqn S21}$$

$$p231 = \frac{r}{\pi}\left(\theta - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S22}$$

1.7.2. *Model p232.* p232 is bounded by $\alpha \leq 3\pi - 2\theta$, $\alpha \geq 2\pi - \theta$ and $\theta \leq \pi$.

p232 does not have the fourth integral from p231 as the right side of the profile does not become viable until after the x_4 region has ended and the x_2 region has begun. Therefore the second



(A)

FIGURE S6. A) If $\alpha/2$, shown in green, is less than $\pi - \theta/2$, as is the case here, then the width of the profile when an animal approaches directly from behind is zero.

x_4 integral has an upper limit of θ and the integral after has a width of $r\cos(x_2 - \theta/2)$ and is integrated with respect to x_2. The final integral starts at $x_4 = 3\pi/2 - \alpha/2 - \theta/2$ and has the full width of $2r\sin(x_2)\sin(\theta/2)$.

$$p232 = \frac{1}{\pi}\left(\int_{\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 + \int_{0}^{-\frac{\pi}{2}+\theta} r - r\cos(-x_4+\theta)\,\mathrm{d}x_4 \right.$$
$$\left. + \int_{-\frac{\pi}{2}+\theta}^{\theta} r\,\mathrm{d}x_4 + \int_{\frac{\theta}{2}}^{\frac{3\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}} r\cos\left(-x_2+\frac{\theta}{2}\right)\mathrm{d}x_2 + \int_{\frac{3\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 \right) \quad \text{eqn S23}$$

$$p232 = \frac{r}{\pi}\left(\theta - \cos\left(\frac{\alpha}{2}\right) + 1\right) \quad \text{eqn S24}$$

1.7.3. *Model p233.* Finally, p233 is bounded by $\alpha \le \pi$, $\theta \ge \pi/2$ and $\alpha \le 3\pi - 2\theta$. p233 is the same as p232 except that the final profile width is zero and this profile is reached at $\alpha/2 + \theta/2 - \pi/2$.

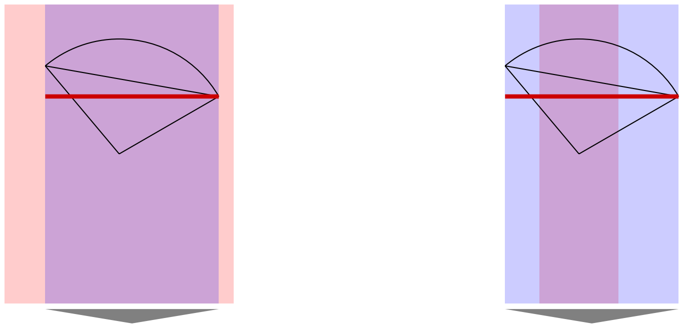
$$p233 = \frac{1}{\pi}\left(\int_{\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 + \int_{0}^{-\frac{\pi}{2}+\theta} r - r\cos(-x_4+\theta)\,\mathrm{d}x_4 \right.$$
$$\left. + \int_{-\frac{\pi}{2}+\theta}^{\theta} r\,\mathrm{d}x_4 + \int_{\frac{\theta}{2}}^{-\frac{\pi}{2}+\frac{\theta}{2}+\frac{\alpha}{2}} r\cos\left(-x_2+\frac{\theta}{2}\right)\mathrm{d}x_2 \right) \quad \text{eqn S25}$$

$$p233 = \frac{r}{\pi}\left(\theta - \cos\left(\frac{\alpha}{2}\right) + 1\right) \quad \text{eqn S26}$$

1.8. **Model p33.** The models in p33 are described with the two focal angles used in models p23, x_2 and x_4. As $\alpha \le \pi$ an animal can never be detected if it is approaching the detector from behind. This makes these models simpler in that they go through the x_2 and x_4 eons only once each.

There are five potential profile sizes. At the beginning of x_2, with an approach direction directly towards the sensor, the factor that limits the width of the profile can either be 1) the sensor width, in which case the profile width is $2r\sin(\theta/2)\sin(x_2)$, or 2) the call width, in which case the profile width is instead $2r\sin(\alpha/2)$ (see Figure S7)

3) The next potential profile in x_2 has a width of $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$ as the right side of the profile is limited by the width of the sensor region while the left side is limited by the call width. However, the angle at which the profile starts depends on whether the first profile was 1) or 2) above. If the first profile is profile 1) then the profile is limited on both sides by the sensor region and then the left side of the profile becomes limited by the call width. This happens at $x_2 = \pi/2 - \alpha/2 + \theta/2$. If however the first profile was 2) then the first profile is limited by the call width. We move into the new profile when the right side of the profile becomes limited by the sensor region. This occurs at $x_2 = \pi/2 + \alpha/2 - \theta/2$.



(A) (B)

FIGURE S7. A) As $\alpha > \theta$ the profile width (purple) is limited by the sensor region, not the call angle (red). The profile width is $2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)$. B) As $\alpha < \theta$ the profile width is limited by the call angle rather than the sensor region (blue). The profile width is $2r\sin\left(\frac{\alpha}{2}\right)$

In the x_4 region the left side of the profile is always $r\sin(\alpha/2)$ while the right side is either 4) 0, giving a profile of $r\sin(\alpha/2)$, or 5) limited by the sensor giving a profile of size $r\sin(\alpha/2) - r\cos(x_4 - \theta)$.

1.8.1. *Model p331.* p331 is bounded by $\alpha \geq \theta$, $\alpha \leq \pi$ and $\theta \leq \pi$.

As α is large the first profile is limited by the size of the sensor region giving it a width of $2r\sin(\theta/2)\sin(x_2)$. It is the only one of the three p33 models to start in this way. Later on, still with x_2 as the focal angle the left side of the profile does become limited by the call width. So at $x_2 = \pi/2 - \alpha/2 + \theta/2$ the profile width becomes $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$.

As we enter the x_4 region, the profile remains limited by the call on the left and by the sensor on the right, giving a profile width of $r\sin(\alpha/2) - r\cos(x_4 - \theta)$. Finally, at $x_4 = \theta - \pi/2$ the right side of the profile becomes zero and the profile is width is $r\sin(\alpha/2)$.

$$
\begin{aligned}
p331 =& \frac{1}{\pi}\left(\int_{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 + \int_{\frac{\theta}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}} -r\cos\left(x_2+\frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_2 \right.\\
&\left. + \int_{0}^{-\frac{\pi}{2}+\theta} -r\cos(-x_4+\theta) + r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_4 + \int_{-\frac{\pi}{2}+\theta}^{-\frac{\pi}{2}+\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_4 \right) \qquad \text{eqn S27}\\
p331 =& \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S28}
\end{aligned}
$$

1.8.2. *Model p332.* p332 is bounded by $\theta \geq \pi/2$, $\alpha \leq \theta$ and $\alpha \geq 2\theta - \pi$.

p332 is largely similar to p331. However, as $\alpha \leq \theta$ the first profile is limited by α and not by the detection region. Therefore the first profile has width $2r\sin(\alpha/2)$. This also means the transition to the second profile occurs at $x_2 = \pi/2 + \alpha/2 - \theta/2$ instead of $x_2 = \pi/2 - \alpha/2 + \theta/2$.

$$
\begin{aligned}
p332 =& \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}+\frac{\alpha}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_2 + \int_{\frac{\theta}{2}}^{\frac{\pi}{2}-\frac{\theta}{2}+\frac{\alpha}{2}} -r\cos\left(x_2+\frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_2 \right.\\
&\left. + \int_{0}^{-\frac{\pi}{2}+\theta} -r\cos(-x_4+\theta) + r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_4 + \int_{-\frac{\pi}{2}+\theta}^{-\frac{\pi}{2}+\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_4 \right) \qquad \text{eqn S29}\\
p332 =& \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S30}
\end{aligned}
$$

1.8.3. *Model p333.* p333 is bounded by $\alpha \leq 2\theta - \pi$ and $\theta \leq \pi$.

p333 is similar to p332 except that the profile does not become limited by sensor at all during the the x_4 regions. Therefore, at $x_4 = 0$ the profile is still of width $2r\sin(\alpha/2)$. Only at $x_4 = \theta - \pi/2 - \alpha/2$ does the profile become limited on the right by the sensor region.

$$p333 = \frac{1}{\pi}\left(\int_{\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_2 + \int_{0}^{-\frac{\pi}{2}+\theta-\frac{\alpha}{2}} 2r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_4 \right.$$
$$\left. + \int_{-\frac{\pi}{2}+\theta-\frac{\alpha}{2}}^{-\frac{\pi}{2}+\theta} -r\cos\left(-x_4+\theta\right) + r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_4 + \int_{-\frac{\pi}{2}+\theta}^{-\frac{\pi}{2}+\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_4 \right) \qquad \text{eqn S31}$$

$$p333 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S32}$$

1.9. **Model p141.** p141 is the model from (Rowcliffe *et al.*, 2008). It has $\alpha = 2\pi$ and $\theta \leq \pi/2$. It has three profile widths, two of which are repeated, once as the animal approaches from on front of the sensor and once as the animal approaches from behind the sensor.

Starting with an approach direction of directly towards the sensor, and examining focal angle x_2, the profile width is $2r\sin(x_2)\sin(\theta/2)$. When the profile is perpendicular to the radius edge of the segment sensor region, we instead examine x_3 where the profile width is $r\sin(x_3)$. At $x_3 = \pi/2$ the profile becomes simply r and this continues for θ radians of x_4. Finally the x_3 and x_2 are repeated with an approach direction from behind the sensor.

$$p141 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin\left(x_2\right) \mathrm{d}x_2 + \int_{\theta}^{\frac{\pi}{2}} r\sin\left(x_3\right) \mathrm{d}x_3 \right.$$
$$\left. + \int_{0}^{\theta} r \,\mathrm{d}x_4 + \int_{\theta}^{\frac{\pi}{2}} r\sin\left(x_3\right) \mathrm{d}x_3 + \int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin\left(x_2\right) \mathrm{d}x_2 \right) \qquad \text{eqn S33}$$

$$p141 = \frac{r}{\pi}\left(\theta + 2\right) \qquad \text{eqn S34}$$

1.10. **Model p24.** In the models in p24, the sensor has $\theta \leq pi/2$ as in p141. As $\alpha \geq \pi/2$ a lot of the profiles are similar to p141. Specifically, the first three profiles are always the same as the first three profiles of p141. This is because when an animal is moving towards the sensor, the $\alpha \geq \pi$ call is no different to a 2π call. However, when approaching the sensor from behind, things are slightly different. The animal can only be detected by the sensor if it's call is wide enough that it can be detected once it has passed the sensor.

The second x_3 profile is always the same width as in p141. This is because there is no detection region to one side of the sensor so this side is unaffected by call width, while the width of the other side of the profile is unaffected by α as when $\alpha > \pi$ the profile width will never be limited by α. If $\alpha \leq 2\pi + 2\theta$, the animal becomes undetectable during this profile when x_3 has decreased in size to $\pi - \alpha/2$. This inequality marks the boundary between p243 and p242.

As the focal angle moves from x_3 to x_2 at $x_3 = \theta$, we can see that if $\alpha \geq 2\pi + 2\theta$, then the x_2 region is reached before the animal become undetectable. When this second x_2 region is reached, the profile starts with width $r\cos(x_2 - \theta/2)$ as at the beginning of the x_2 profile as only animals approaching to the left of the sensor are detectable. The sensor is directly behind the right side of the profile.

During this second x_2 profile the call angle needed for animals to be detected to the left of the detector is increasing while the angle needed for animals to be detected to the right of the detector is decreasing. Therefore, either the left side becomes undetectable, making both sides undetectable (this occurs if $\alpha \leq 2\pi - \theta$ as in p242) or the right becomes detectable (if $\alpha \geq 2\pi - \theta$ as in p241), making both sides detectable and giving a profile width of $2r\sin(x_2)\sin(\theta/2)$.

1.10.1. *Model p241.* p241 is bounded by $\alpha \geq 2\pi - \theta$, $\alpha \leq 2\pi$ and $\theta \leq \pi/2$.

It is the same as p141 except that it includes the extra profile in x_2 (the fifth integral) where only animals approaching to the left of the profile are detected.

$$p241 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 + \int_{\theta}^{\frac{\pi}{2}} r\sin(x_3)\,\mathrm{d}x_3 + \int_{0}^{\theta} r\,\mathrm{d}x_4 \right.$$
$$\left. + \int_{\theta}^{\frac{\pi}{2}} r\sin(x_3)\,\mathrm{d}x_3 + \int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{3\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}} r\cos\left(-x_2+\frac{\theta}{2}\right)\mathrm{d}x_2 + \int_{\frac{3\pi}{2}-\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 \right) \qquad \text{eqn S35}$$

$$p241 = \frac{r}{\pi}\left(\theta - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S36}$$

1.10.2. *Model p242.* p242 is bounded by $\alpha \leq 2\pi - \theta$, $\alpha \geq 2\pi + 2\theta$ and $\theta \leq \pi/2$

p242 is the same p241 except that as $\alpha \leq 2\pi - \theta$, animals that approach from directly behind the detector are not detected. Therefore at $x_2 = \alpha/2 + \theta/2 - \pi/2$ the profile width goes to zero and therefore the last integral in p241 is not included.

$$p242 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 + \int_{\theta}^{\frac{\pi}{2}} r\sin(x_3)\,\mathrm{d}x_3 \right.$$
$$\left. + \int_{0}^{\theta} r\,\mathrm{d}x_4 + \int_{\theta}^{\frac{\pi}{2}} r\sin(x_3)\,\mathrm{d}x_3 + \int_{\frac{\pi}{2}-\frac{\theta}{2}}^{-\frac{\pi}{2}+\frac{\theta}{2}+\frac{\alpha}{2}} r\cos\left(-x_2+\frac{\theta}{2}\right)\mathrm{d}x_2 \right) \qquad \text{eqn S37}$$

$$p242 = \frac{r}{\pi}\left(\theta - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S38}$$

1.10.3. *Model p243.* p243 is bounded by $\alpha \geq 2\pi + 2\theta$, $\alpha \geq \pi$ and $\theta \geq 0$.

It is similar to p242 but doesn't include the last integral as during the x_3 profile, at $x_3 = \pi - \alpha/2$ the call width is too small for any animals to be detected, so the profile width goes to zero.

$$p243 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\,\mathrm{d}x_2 + \int_{\theta}^{\frac{\pi}{2}} r\sin(x_3)\,\mathrm{d}x_3 \right.$$
$$\left. + \int_{0}^{\theta} r\,\mathrm{d}x_4 + \int_{\pi-\frac{\alpha}{2}}^{\frac{\pi}{2}} r\sin(x_3)\,\mathrm{d}x_3 \right) \qquad \text{eqn S39}$$

$$p243 = \frac{r}{\pi}\left(\theta - \cos\left(\frac{\alpha}{2}\right) + 1\right) \qquad \text{eqn S40}$$

1.11. **Model p34.** Cell p34 is split into six models rather than three like most of the other cells. As $\alpha < \pi$, animals approaching the sensor from behind can never be detected, so unlike p141, the second x_2 and x_3 profiles are always zero. The six models are split by three inequalities that relate to the models as follows.

Models with $\alpha \leq \pi - 2\theta$ have no x_4 profile. This is because at $x_4 = 0$, the call angle is already too small to be detected as can be seen in Figure S8a where $\alpha/2 < \pi/2 - \theta$ which simplifies to give the previous inequality.

Models with $\alpha \leq \theta$ are limited by α in the first, x_2 region (see Figure S7), rather than being limited by θ. Therefore this first profile is of width $2r\sin(\alpha/2)$ rather than $2r\sin(\theta/2)\sin(x_2)$.

Finally, models with $\alpha \leq 2\theta$ have a second profile in x_2 where to one side of the sensor α is the limiting factor of profile width, while on the other side θ is (see Figure S8b). This gives a width of $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$. This profile doesn't occur in models with $\alpha \geq 2\theta$.

1.11.1. *Model p341.* p341 is bounded by $\alpha \leq \theta$, $\alpha \geq \pi - 2\theta$ and $\theta \leq \pi/2$. Therefore it does contain a x_4 profile, starts with an α limited profile and does contain the $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$ profile in x_2.

(A) $x_4 = 0$ (B)

FIGURE S8. A) At $x_4 = 0$, if $\alpha < \pi - 2\theta$ then $\alpha/2$ is too small for an animal to be detected at all during the x_4 profile. B) The left of the profile is limited by the call width, not the sensor (blue). On the right, the profile is limited by the sensor and not the call (red). Overall the profile width is $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$.

$$p341 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}+\frac{\alpha}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_2 + \int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}-\frac{\theta}{2}+\frac{\alpha}{2}} -r\cos\left(x_2 + \frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_2 + \int_{\theta}^{\frac{\pi}{2}} r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_3 + \int_{0}^{-\frac{\pi}{2}+\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_4 \right) \quad \text{eqn S41}$$

$$p341 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \quad \text{eqn S42}$$

1.11.2. *Model p342.* p342 is the only model with a tetrahedral bounding region. It is bounded by $\alpha \geq \theta$, $\alpha \geq \pi - 2\theta$, $\alpha \leq 2\theta$ and $\theta \leq \pi/2$. Therefore it does contain a x_4 profile, but starts with a θ limited profile. It does contain the $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$ profile in x_2.

$$p342 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2) \mathrm{d}x_2 + \int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}} -r\cos\left(x_2 + \frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_2 + \int_{\theta}^{\frac{\pi}{2}} r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_3 + \int_{0}^{-\frac{\pi}{2}+\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_4 \right) \quad \text{eqn S43}$$

$$p342 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \quad \text{eqn S44}$$

1.11.3. *Model p343.* p343 is bounded by $\alpha \geq \pi - 2\theta$, $\alpha \geq 2\theta$ and $\alpha \leq \pi$. It starts with a θ limeted profile and has a x_4 profile. However, it does not contain the $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$ profile.

$$p343 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2) \mathrm{d}x_2 + \int_{\theta}^{\frac{\alpha}{2}} r\sin(x_3) \mathrm{d}x_3 + \int_{\frac{\alpha}{2}}^{\frac{\pi}{2}} r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_3 + \int_{0}^{-\frac{\pi}{2}+\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_4 \right) \quad \text{eqn S45}$$

$$p343 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \quad \text{eqn S46}$$

1.11.4. *Model p344.* p344 is bounded by $\alpha \leq \pi - 2\theta$, $\alpha \leq \theta$ and $\alpha < 0$. Therefore it does not contain a x_4 profile. It starts with an α limited profile and contains the $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$ profile in x_2.

$$p344 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}+\frac{\alpha}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\alpha}{2}\right) \mathrm{d}x_2 + \int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}-\frac{\theta}{2}+\frac{\alpha}{2}} -r\cos\left(x_2+\frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_2 + \int_{\theta}^{\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_3 \right) \quad \text{eqn S47}$$

$$p344 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \quad \text{eqn S48}$$

1.11.5. *Model p345.* p345 is bounded by $\alpha \leq \pi - 2\theta$, $\alpha \geq \theta$ and $alpha \leq 2\theta$. It starts with a θ limited profile. It does contain the $r\sin(\alpha/2) - r\cos(x_2 + \theta/2)$ profile in x_2 but does not have a x_4 profile.

$$p345 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\, \mathrm{d}x_2 + \int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}+\frac{\theta}{2}-\frac{\alpha}{2}} -r\cos\left(x_2+\frac{\theta}{2}\right) + r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_2 + \int_{\theta}^{\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_3 \right) \quad \text{eqn S49}$$

$$p345 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \quad \text{eqn S50}$$

1.11.6. *Model p346.* Finally, p346, the last model, is bounded by y $\alpha \leq \pi - 2\theta$, $alpha \geq 2\theta$ and $\theta \geq 0$. Therefore it starts with a θ limited profile. However it doesn't contain the extra x_2 profile nor a x_4 profile.

$$p346 = \frac{1}{\pi}\left(\int_{\frac{\pi}{2}-\frac{\theta}{2}}^{\frac{\pi}{2}} 2r\sin\left(\frac{\theta}{2}\right)\sin(x_2)\, \mathrm{d}x_2 + \int_{\theta}^{\frac{\alpha}{2}} r\sin(x_3)\, \mathrm{d}x_3 + \int_{\frac{\alpha}{2}}^{\theta+\frac{\alpha}{2}} r\sin\left(\frac{\alpha}{2}\right)\mathrm{d}x_3 \right) \quad \text{eqn S51}$$

$$p346 = \frac{r}{\pi}\left(\theta\sin\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\alpha}{2}\right) + 1\right) \quad \text{eqn S52}$$

2. Supplementary Script: Symbolic algebra Python Script

This script uses the SymPy package SymPy Development Team (2014), a computer algebra system to calculate the equations for p in the various models and to perform unit checks on the results.

```
"""
Systematic analysis of REM models
Tim Lucas
01/10/13
"""


from sympy import *
import numpy as np
import matplotlib.pyplot as pl
from datetime import datetime
import Image as Im


# Use LaTeX printing
from sympy import init_printing ;
init_printing()
# Make LaTeX output white. Because I use a dark theme
init_printing(forecolor="White")


# Load symbols used for symbolic maths
t, a, r, x2, x3, x4, x1 = symbols('theta alpha r x_2 x_3 x_4 x_1', positive=True)
r1 = {r:1} # useful for lots of checks


# Define functions to neaten up later code.

# Calculate the final profile averaged over pi.
def calcModel(model):
        x = pi**-1 * sum( [integrate(m[0], m[1:]) for m in model] ).simplify().trigsimp()
        return x

# Do the replacements fit within the area defined by the conditions?
def confirmReplacements(conds, reps):
        if not all([c.subs(reps) for c in eval(conds)]):
                print('reps' + conds[4:] + ' incorrect')

# is average profile in range 0r-2r?
def profileRange(prof, reps):
        if not 0 <= eval(prof).subs(dict(reps, **r1)) <= 2:
                print('Total ' + prof + ' not in 0, 2r')

# Are the individuals integrals >0r
def intsPositive(model, reps):
        m = eval(model)
        for i in range(len(m)):
                if not integrate(m[i][0], m[i][1:]).subs(dict(reps, **r1)) > 0:
                    print('Integral ' + str(i+1) + ' in ' + model + ' is negative')

# Are the individual averaged integrals between 0 and  2r
def intsRange(model, reps):
        m = eval(model)
        for i in range(len(m)):
                if not 0 <= (integrate(m[i][0], m[i][1:])/(m[i][3]-m[i][2])).subs(dict(reps, **r1)) <= 2:
                        print('Integral ' + str(i+1) + ' in ' + model + ' has averaged integral outside 0<p<2r')

# Are the bounds the correct way around
def checkBounds(model, reps):
        m = eval(model)
        for i in range(len(m)):
                if not (m[i][3]-m[i][2]).subs(reps) > 0:
                        print('Bounds ' + str(i+1) + ' in ' + model + ' has lower bounds bigger than upper bounds')

# create latex strings with the 1) the integral equation that defines it and 2)  the final calculated model.
# There's some if statements to split longer equations on two lines and get +s in the right place.
def parseLaTeX(prof):
        m = eval( 'm' + prof[1:] )
        f = open('/home/tim/Dropbox/PhD/Analysis/REM-chapter/latexFiles/'+prof+'.tex', 'w')
        f.write('\\begin{align}\n    ' + prof + ' =&\\frac{1}{\pi} \left(\;\;')
        for i in range(len(m)):
                f.write('\int\limits_{'+latex(m[i][2], order='rev-lex')+'}^{'+latex(m[i][3], order='rev-lex')+'}'+latex(m[i][0], order='rev-lex')+'\;\mathrm{d}' +latex(m[i][1]))
                if len(m)>3 and i==(len(m)/2)-1:
                        f.write( '\\right.\\notag\\\\\n &\left.' )
                if i<len(m)-1:
                        f.write('+')
        f.write('\\right)\label{' + prof + 'Def}\\\\\n    ')
```

```
        f.write(prof + ' =& ' + latex(eval(prof)) + '\label{' + prof + 'Sln}\n\\end{align}')
        f.close()


# Apply all checks.
def allChecks(prof):
        model = 'm' + prof[1:]
        reps = eval('rep' + prof[1:])
        conds = 'cond' + prof[1:]
        confirmReplacements(conds, reps)
        profileRange(prof, reps)
        intsPositive(model, reps)
        intsRange(model, reps)
        checkBounds(model, reps)

##########################################################
# 221 animal: a = 2*pi.  sensor: t > pi, a > 3pi - t  #
##########################################################



m221 = [ [2*r,                 x1, pi/2, t/2        ],
         [r + r*cos(x1 - t/2), x1, t/2,  pi         ],
         [r + r*cos(x1 + t/2), x1, pi,   2*pi-t/2   ],
         [2*r,                 x1, 2*pi-t/2, 3*pi/2 ] ]

# Replacement values in range
rep221 = {t:3*pi/2, a:2*pi}

# Define conditions for model
cond221 = [pi <= t, a >= 3*pi - t]

# Calculate model, run checks, write output.
p221 = calcModel(m221)
allChecks('p221')
parseLaTeX('p221')

################################################################################
# 222 animal: a > pi.  sensor: t > pi Condition: a < 3pi - t, a > 4pi - 2t  #
################################################################################



m222 = [ [2*r,                 x1, pi/2, t/2        ],
         [r + r*cos(x1 - t/2), x1, t/2,  5*pi/2 - t/2 - a/2 ],
         [r + r*cos(x1 + t/2), x1, 5*pi/2 - t/2 - a/2,   2*pi-t/2 ],
         [2*r,                 x1, 2*pi-t/2, 3*pi/2 ] ]


# Replacement values in range
rep222 = {t:5*pi/3, a:4*pi/3-0.1}

# Define conditions for model
cond222 = [pi <= t, a >= pi, a <= 3*pi - t, a >= 4*pi - 2*t]

# Calculate model, run checks, write output.
p222 = calcModel(m222)
allChecks('p222')
parseLaTeX('p222')


###################################################################
# 223 animal: a > pi.  sensor: t > pi Condition: a < 4pi - 2t  #
###################################################################



m223 = [ [2*r,                 x1, pi/2, t/2        ],
         [r + r*cos(x1 - t/2), x1, t/2,  t/2 + pi/2         ],
         [r                  , x1, t/2 + pi/2,   5*pi/2 - t/2 - a/2 ],
         [r + r*cos(x1 + t/2), x1, 5*pi/2 - t/2 - a/2,   2*pi-t/2 ],
         [2*r,                 x1, 2*pi-t/2, 3*pi/2 ] ]


# Replacement values in range
rep223 = {t:5*pi/4-0.1, a:3*pi/2}

# Define conditions for model
cond223 = [pi <= t, a >= pi, a <= 4*pi - 2*t]

# Calculate model, run checks, write output.
p223 = calcModel(m223)
allChecks('p223')
parseLaTeX('p223')
```

```
##########################################################
# 131 animal: a = 2*pi.   sensor:  pi/2 <= t <= pi      #
##########################################################

m131 = [ [2*r*sin(t/2)*sin(x2), x2, t/2,      pi/2     ],
        [r - r*cos(x4 - t),      x4, 0,        t - pi/2 ],
        [r,                      x4, t - pi/2, pi/2     ],
        [r - r*cos(x4),          x4, pi/2,     t        ],
        [2*r*sin(t/2)*sin(x2),   x2, t/2,      pi/2     ] ]

# Replacement values in range
rep131 = {t:3*pi/4}

# Define conditions for model
cond131 = [pi/2 <= t, t <= pi]

# Calculate model, run checks, write output.
p131 = calcModel(m131)
allChecks('p131')
parseLaTeX('p131')



####################################################################################
# 231 animal: a > pi.  Sensor: pi/2 <= t <= pi. Condition: a > 2pi - t            #
####################################################################################


m231 = [ [2*r*sin(t/2)*sin(x2), x2, t/2,          pi/2        ],
         [r - r*cos(x4 - t),    x4, 0,            t - pi/2    ],
         [r,                    x4, t - pi/2,     3*pi/2 - a/2],
         [r - r*cos(x4),        x4, 3*pi/2 - a/2, t           ],
         [2*r*sin(t/2)*sin(x2), x2, t/2,          pi/2        ] ]


rep231 = {t:3*pi/4, a:15*pi/8} # Replacement values in range

# Define conditions for model
cond231 = [a > pi, pi/2 <= t, t <= pi, a >= 3*pi - 2*t]

# Calculate model, run checks, write output.
p231 = calcModel(m231)
allChecks('p231')
parseLaTeX('p231')


####################################################################################
# 232 animal: a > pi.  Sensor: pi/2 <= t <= pi. Cond: 2pi - t < a < 3pi - 2t     #
####################################################################################


m232 = [ [2*r*sin(t/2)*sin(x2), x2, t/2,                pi/2              ],
         [r - r*cos(x4 - t),    x4, 0,                  t - pi/2          ],
         [r,                    x4, t - pi/2,           t                 ],
         [r*cos(x2 - t/2),      x2, t/2,                3*pi/2 - a/2 - t/2],
         [2*r*sin(t/2)*sin(x2), x2, 3*pi/2 - a/2 - t/2, pi/2              ] ]


rep232 = {t:5*pi/8, a:6*pi/4} # Replacement values in range

# Define conditions for model
cond232 = [a > pi, pi/2 <= t, t <= pi, 2*pi - t <= a, a <= 3*pi - 2*t]

# Calculate model, run checks, write output.
p232 = calcModel(m232)
allChecks('p232')
parseLaTeX('p232')


####################################################################################
# 233 animal:  a > pi.  Sensor: pi/2 <= t <= pi. Condition: a <= 2pi - t         #
####################################################################################

m233 = [ [2*r*sin(t/2)*sin(x2), x2, t/2, pi/2],
         [r - r*cos(x4 - t),    x4, 0, t - pi/2],
         [r,                    x4, t - pi/2, t],
         [r*cos(x2 - t/2),      x2, t/2, a/2 + t/2 - pi/2] ]

rep233 = {t:3*pi/4, a:9*pi/8} # Replacement values in range

# Define conditions for model
cond233 = [a > pi,  pi/2 <= t, t <= pi, a <= 2*pi - t]

# Calculate model, run checks, write output.
p233 = calcModel(m233)
allChecks('p233')
```

```
parseLaTeX('p233')

##############################################################################
# 141 animal: a=2pi.  Sensor: t <= pi/2.                                     #
##############################################################################

m141 = [ [2*r*sin(t/2)*sin(x2), x2, pi/2 - t/2, pi/2],
         [r*sin(x3),            x3, t,          pi/2],
         [r,                    x4, 0*t,          t],
         [r*sin(x3),            x3, t,          pi/2],
         [2*r*sin(t/2)*sin(x2), x2, pi/2 - t/2, pi/2] ]


rep141 = {t:3*pi/8, a:2*pi} # Replacement values in range

# Define conditions for model
cond141 = [ t <= pi/2  ]

# Calculate model, run checks, write output.
p141 = calcModel(m141)
allChecks('p141')
parseLaTeX('p141')


##############################################################################
# 241 animal: a>pi.  Sensor: t <= pi/2. Condition: 2*pi - t < a              #
##############################################################################

m241 = [ [2*r*sin(t/2)*sin(x2), x2, pi/2 - t/2, pi/2],
         [r*sin(x3),            x3, t,          pi/2],
         [r,                    x4, 0,          t],
         [r*sin(x3),            x3, t,          pi/2],
         [r*cos(x2 - t/2),      x2, pi/2 - t/2, 3*pi/2 - t/2 - a/2],
         [2*r*sin(t/2)*sin(x2), x2, 3*pi/2 - t/2 - a/2, pi/2] ]


rep241 = {t:3*pi/8, a:29*pi/16} # Replacement values in range

# Define conditions for model
cond241 = [a >= pi, t <= pi/2, 2*pi - t <= a  ]

# Calculate model, run checks, write output.
p241 = calcModel(m241)
allChecks('p241')
parseLaTeX('p241')

####################################################################################
# 242 animal: a>pi.  Sensor: t <= pi/2. Condition:  2*pi - 2*t <= a <= 2*pi - t  #
####################################################################################

m242 = [ [2*r*sin(t/2)*sin(x2), x2, pi/2 - t/2, pi/2],
         [r*sin(x3),            x3, t,          pi/2],
         [r,                    x4, 0,          t],
         [r*sin(x3),            x3, t,          pi/2],
         [r*cos(x2 - t/2),      x2, pi/2 - t/2, a/2 + t/2 - pi/2] ]

rep242 = {t:3*pi/8, a:3*pi/2} # Replacement values in range

# Define conditions for model
cond242 = [a >= pi, t <= pi/2, 2*pi - 2*t <= a, a <= 2*pi - t]

# Calculate model, run checks, write output.
p242 = calcModel(m242)
allChecks('p242')
parseLaTeX('p242')


######################################################################
# 243 animal: a>pi.  Sensor: t <= pi/2. Condition: a <= 2pi - 2t  #
######################################################################

m243 = [ [2*r*sin(t/2)*sin(x2), x2, pi/2 - t/2, pi/2],
         [r*sin(x3),            x3, t,          pi/2],
         [r,                    x4, 0,          t   ],
         [r*sin(x3),            x3, pi - a/2,   pi/2] ]


rep243 = {t:pi/9, a:10*pi/9} # Replacement values in range

# Define conditions for model
cond243 = [t <= pi/2, a >= pi, a <= 2*pi - 2*t]

# Calculate model, run checks, write output.
p243 = calcModel(m243)
allChecks('p243')
parseLaTeX('p243')
```

```
#####################################################
# 311 animal: a <= pi.  Sensor: t =2pi.             #
#####################################################


m311 = [ [ 2*r*sin(a/2),                        x1, pi/2, 3*pi/2       ],
         ]


rep311 = {a:pi/4} # Replacement values in range

# Define conditions for model
cond311 = [a <= pi]

# Calculate model, run checks, write output.
p311 = calcModel(m311)
allChecks('p311')
parseLaTeX('p311')



###################################################################################
# 321 animal: a <= pi.  Sensor: t > pi. Condition: a > 2pi - t, a > 4pi - 2t    #
###################################################################################


m321 = [ [ 2*r*sin(a/2),                        x1, pi/2,               t/2 + pi/2 - a/2       ],
         [ r*sin(a/2) + r*cos(x1 - t/2),        x1, t/2 + pi/2 - a/2,   5*pi/2 - a/2 - t/2 ],
         [ 2*r*sin(a/2),                        x1, 5*pi/2 - a/2 - t/2, 3*pi/2]  ]


rep321 = {t:19*pi/10, a:pi/2} # Replacement values in range

# Define conditions for model
cond321 = [a <= pi, t >= pi,  a >= 4*pi - 2*t]

# Calculate model, run checks, write output.
p321 = calcModel(m321)
allChecks('p321')
parseLaTeX('p321')



###########################################################################################
# 322 animal: a <= pi.  Sensor: t > pi. Condition: 2pi - t < a < 4pi - 2t #
###########################################################################################

m322 = [ [ 2*r*sin(a/2),                        x1, pi/2,               t/2 + pi/2 - a/2  ],
         [ r*sin(a/2) + r*cos(x1 - t/2),        x1, t/2 + pi/2 - a/2,   t/2 + pi/2        ],
         [ r*sin(a/2),                          x1, t/2 + pi/2,         5*pi/2 - a/2 - t/2],
         [ 2*r*sin(a/2),                        x1, 5*pi/2 - a/2 - t/2, 3*pi/2            ] ]

rep322 = {t:3*pi/2 + 0.1, a:pi/2} # Replacement values in range

# Define conditions for model
cond322 = [a <= pi, t >= pi,  a >= 2*pi - t, a <= 4*pi - 2*t]

# Calculate model, run checks, write output.
p322 = calcModel(m322)
allChecks('p322')
parseLaTeX('p322')



#####################################################################################
# 323 animal: a <= pi.  Sensor: t > pi. Condition: a <= 4*pi - 2*t and a < 2*pi - t #
#####################################################################################

m323 = [ [ 2*r*sin(a/2),                       x1, pi/2,             t/2 + pi/2 - a/2  ],
         [ r*sin(a/2) + r*cos(x1 - t/2),       x1, t/2 + pi/2 - a/2, t/2 + pi/2        ],
         [ r*sin(a/2),                         x1, t/2 + pi/2,       t/2 + pi/2 + a/2  ] ]


rep323 = {t:3*pi/2, a:pi/3} # Replacement values in range


# Define conditions for model
cond323 = [a <= pi, t >= pi/2, a <= 4*pi - 2*t , a <= 2*pi - t]

# Calculate model, run checks, write output.
p323 = calcModel(m323)
allChecks('p323')
parseLaTeX('p323')


###################################################################################

```

```
"""
Ccomplex profiles for a <= pi/2
These were specified using a very roundabout way that I realised isn't necessary.
Worth keeping them here just for the record.

# p-l-r for x2 profil. Calculated by AE in fig 22.4 minus AE in fig 22.3
p1 = (2*r*sin(t/4 - x2/2 + pi/4 + a/4)*sin(a/4 + pi/4 + x2/2 - t/4) - \
     2*r*sin((pi - a - 2*x2 + t)/4)*sin((pi - a + 2*x2 - t)/4)).simplify()

# p-l for x2 profiles
p2 = (2*r*sin(t/2)*sin(x2) - 2*r*sin((pi - a - 2*x2 + t)/4)*sin((pi - a + 2*x2 - t)/4)).simplify()

# p-l for x3 profile.
p3 = (r*sin(x3) - (2*r*sin(x3/2 - a/4)*sin(pi/2 - x3/2 - a/4)).simplify()).trigsimp()
"""


##############################################################################################
# 331 animal: a <= pi.  Sensor: pi/2 <= t <= pi. Condition: a >= t and a/2 >= t - pi/2 #
##############################################################################################


m331 =  [ [2*r*sin(t/2)*sin(x2),              x2, pi/2 - a/2 + t/2, pi/2             ],
          [r*sin(a/2) - r*cos(x2 + t/2),      x2, t/2,              pi/2 - a/2 + t/2],
          [r*sin(a/2) - r*cos(x4 - t),        x4, 0,                t - pi/2         ],
          [r*sin(a/2),                        x4, t-pi/2,           t - pi/2 + a/2   ] ]


rep331 = {t:5*pi/8, a:6*pi/8} # Replacement values in range

# Define conditions for model
cond331 = [a <= pi, pi/2 <= t, t <= pi, a >= t, a/2 >= t - pi/2]

# Calculate model, run checks, write output.
p331 = calcModel(m331)
allChecks('p331')
parseLaTeX('p331')




#############################################################################################
# 332 animal: a <= pi.  Sensor: pi/2 <= t <= pi. Condition: a <= t and a/2 >= t- pi/2 #
#############################################################################################


m332 =  [ [2*r*sin(a/2),                 x2, pi/2 + a/2 - t/2, pi/2             ],
          [r*sin(a/2) - r*cos(x2 + t/2), x2, t/2,              pi/2 + a/2 - t/2],
          [r*sin(a/2) - r*cos(x4 - t),   x4, 0*t,              t - pi/2       ],
          [r*sin(a/2),                   x4, t - pi/2,         t - pi/2 + a/2 ] ]


rep332 = {t:7*pi/8, a:7*pi/8-0.1} # Replacement values in range

# Define conditions for model
cond332 = [a <= pi, pi/2 <= t, t <= pi, a/2 <= t/2, a/2 >= t - pi/2]

# Calculate model, run checks, write output.
p332 = calcModel(m332)
allChecks('p332')
parseLaTeX('p332')







#############################################################################################
# 333 animal: a <= pi.  Sensor: pi/2 <= t <= pi. Condition: a <= t and a/2 <= t- pi/2 #
#############################################################################################



m333 =  [ [2*r*sin(a/2),                      x2, t/2,              pi/2           ],
          [2*r*sin(a/2),                      x4, 0,                t - pi/2 - a/2 ],
          [r*sin(a/2) - r*cos(x4 - t),        x4, t - pi/2 - a/2,   t - pi/2       ],
          [r*sin(a/2),                        x4, t - pi/2,         t - pi/2 + a/2 ] ]


rep333 = {t:7*pi/8, a:2*pi/8} # Replacement values in range

# Define conditions for model
cond333 = [a <= pi, pi/2 <= t, t <= pi, a/2 <= t/2, a/2 <= t - pi/2]

# Calculate model, run checks, write output.
```

```
p333 = calcModel(m333)
allChecks('p333')
parseLaTeX('p333')




################################################################################
# 341 animal: a <= pi.  Sensor: t <= pi/2. Condition: a > pi - 2t &  a <= t      #
################################################################################


m341 = [ [2*r*sin(a/2),               x2, pi/2 - t/2 + a/2, pi/2            ],
         [r*sin(a/2) - r*cos(x2 + t/2), x2, pi/2 - t/2,       pi/2 - t/2 + a/2],
         [r*sin(a/2),                 x3, t,                pi/2            ],
         [r*sin(a/2),                 x4, 0,                a/2 + t - pi/2  ] ]

rep341 = {t:pi/2-0.1, a:pi/4} # Replacement values in range

# Define conditions for model
cond341 = [a <= pi,  t <= pi/2,  a >= pi - 2*t,  a <= t]

# Calculate model, run checks, write output.
p341 = calcModel(m341)
allChecks('p341')
parseLaTeX('p341')


####################################################################################
# 342 animal: a <= pi.  Sensor: t <= pi/2. Condition: a > pi - 2t &  t <= a <= 2t    #
####################################################################################


m342 = [ [2*r*sin(t/2)*sin(x2),         x2, pi/2 + t/2 - a/2, pi/2            ],
         [r*sin(a/2) - r*cos(x2 + t/2), x2, pi/2 - t/2,       pi/2 + t/2 - a/2],
         [r*sin(a/2),                 x3, t,                pi/2        ],
         [r*sin(a/2),                 x4, 0,                a/2 + t -pi/2   ] ]


rep342 = {t:pi/2-0.1, a:pi/2} # Replacement values in range

# define conditions for model
cond342 = [a <= pi,  t <= pi/2,  a >= pi - 2*t,  t <= a, a <= 2*t]


# Calculate model, run checks, write output.
p342 = calcModel(m342)
allChecks('p342')
parseLaTeX('p342')




################################################################################
# 343 animal: a <= pi.  Sensor: t <= pi/2. Condition: a > pi - 2t &  a > 2t      #
################################################################################



m343 = [ [2*r*sin(t/2)*sin(x2), x2, pi/2 - t/2, pi/2            ],
         [r*sin(x3),            x3, t,          a/2             ],
         [r*sin(a/2),           x3, a/2,        pi/2            ],
         [r*sin(a/2),           x4, 0,          a/2 + t -pi/2   ] ]


rep343 = {t:pi/4, a:3*pi/4} # Replacement values in range


# Define conditions for model
cond343 = [a <= pi,  t <= pi/2,  a >= pi - 2*t,  a > 2*t]

# Calculate model, run checks, write output.
p343 = calcModel(m343)
allChecks('p343')
parseLaTeX('p343')




#################################################################################
# 344 animal: a <= pi.  Sensor: t <= pi/2. Condition: a <= pi - 2t & a <= t      #
#################################################################################

m344 = [ [2*r*sin(a/2),               x2, pi/2 - t/2 + a/2, pi/2            ],
         [r*sin(a/2) - r*cos(x2 + t/2), x2, pi/2 - t/2,       pi/2 - t/2 + a/2],
```

```
         [r*sin(a/2),                   x3, t,                t + a/2         ] ]


rep344 = {t:2*pi/8, a:pi/8} # Replacement values in range

# Define conditions for model
cond344 = [a <= pi, t <= pi/2, a <= pi - 2*t, a <= t]

# Calculate model, run checks, write output.
p344 = calcModel(m344)
allChecks('p344')
parseLaTeX('p344')




######################################################################################
# 345 animal: a <= pi.  Sensor: t <= pi/2. Condition: a <= pi - 2t & t <= a <= 2t   #
######################################################################################


m345 = [ [2*r*sin(t/2)*sin(x2),         x2, pi/2 + t/2 - a/2, pi/2            ],
         [r*sin(a/2) - r*cos(x2 + t/2), x2, pi/2 - t/2,       pi/2 + t/2 - a/2],
         [r*sin(a/2),                   x3, t,                t + a/2         ] ]

rep345 = {t:2*pi/8, a:pi/2-0.1} # Replacement values in range

# Define conditions for model
cond345 = [a <= pi, t <= pi/2, a <= pi - 2*t, t <= a, a <= 2*t]

# Calculate model, run checks, write output.
p345 = calcModel(m345)
allChecks('p345')
parseLaTeX('p345')



#################################################################################
# 346 animal: a <= pi.  Sensor: t <= pi/2. Condition: a <= pi - 2t &  2t <= a      #
#################################################################################

m346 = [ [2*r*sin(t/2)*sin(x2), x2, pi/2 - t/2, pi/2    ],
         [r*sin(x3),            x3, t,          a/2     ],
         [r*sin(a/2),           x3, a/2,        t + a/2 ] ]


rep346 = {t:1*pi/8, a:pi/2} # Replacement values in range

# Define conditions for model
cond346 = [a <= pi,  t <= pi/2,  a <= pi - 2*t,  2*t <= a]

# Calculate model, run checks, write output.
p346 = calcModel(m346)
allChecks('p346')
parseLaTeX('p346')




#
    ################################################################################################################


####################
## Run tests     ###
####################

# create gas model object
gas = 2*r


# for each model run through every adjacent model.
# Contains duplicatea but better for avoiding missed comparisons.
# Also contains replacement t->a and a->t just in case.


allComps = [
['gas', 'p221', {t:2*pi}],
['gas', 'p311', {a:pi}],

['p221', 'gas', {t:2*pi}],
['p221', 'p131', {t:pi}],
['p221', 'p222',{a:3*pi-t}],
['p221', 'p222',{t:3*pi-a}],

['p222', 'p221',{a:3*pi-t}],
['p222', 'p221',{t:3*pi-a}],
```

```
['p222', 'p223',{a:4*pi-2*t}],
['p222', 'p223',{t:2*pi-a/2}],
['p222', 'p321',{a:pi}],

['p223', 'p222',{a:4*pi-2*t}],
['p223', 'p222',{t:2*pi-a/2}],
['p223', 'p322',{a:pi}],
['p223', 'p231',{t:pi}],

['p131','p221', {t:pi}],
['p131','p231',{a:2*pi}],

['p231','p223',{t:pi}],
['p231','p232',{a:3*pi-2*t}],
['p231','p232',{t:3*pi/2-a/2}],
['p231','p131',{a:2*pi}],

['p232','p241',{t:pi/2}],
['p232','p233',{a:2*pi-t}],
['p232','p233',{t:2*pi-a}],
['p232','p231',{a:3*pi-2*t}],
['p232','p231',{t:3*pi/2-a/2}],

['p233','p242',{t:pi/2}],
['p233','p232',{t:2*pi-a}],
['p233','p232',{a:2*pi-t}],
['p233','p331',{a:pi}],

['p141','p131', {t:pi/2}],
['p141','p241',{a:2*pi}],

['p241','p141',{a:2*pi}],
['p241','p242',{a:2*pi-t}],
['p241','p242',{t:2*pi-a}],
['p241','p232',{t:pi/2}],

['p242','p241',{a:2*pi-t}],
['p242','p241',{t:2*pi-a}],
['p242','p243',{t:pi-a/2}],
['p242','p243',{a:2*pi-2*t}],
['p241','p233',{t:pi/2}],

['p243','p242',{t:2*pi-2*a}],
['p243','p242',{a:2*pi-2*t}],
['p243','p343',{a:pi}],

['p311','p321',{t:2*pi}],
['p311','gas',{a:pi}],

['p321','p322',{t:2*pi-a/2}],
['p321','p322',{a:4*pi-2*t}],
['p321','p311',{t:2*pi}],
['p321','p222',{a:pi}],

['p322','p321',{a:4*pi-2*t}],
['p322','p321',{t:2*pi-a/2}],
['p322','p323',{a:2*pi-t}],
['p322','p323',{t:2*pi-a}],
['p322','p223',{a:pi}],

['p323','p322',{t:2*pi-a}],
['p323','p322',{a:2*pi-t}],
['p323','p333',{t:pi}],

['p331','p342',{t:pi/2}],
['p331','p332',{a:t}],
['p331','p332',{t:a}],
['p331','p233',{a:pi}],

['p332','p331',{a:t}],
['p332','p331',{t:a}],
['p332','p341',{t:pi/2}],
['p332','p333',{a:2*t-pi}],
['p332','p333',{t:a/2+pi/2}],

['p333','p332',{t:a/2+pi/2}],
['p333','p332',{a:2*t-pi}],
['p333','p323',{t:pi}],


['p341','p344',{a:pi-2*t}],
['p341','p344',{t:pi/2-a/2}],
['p341','p342',{t:a}],
['p341','p342',{a:t}],
['p341','p332',{t:pi/2}],

['p342','p341',{t:a}],
```

```
['p342','p341',{a:t}],
['p342','p345',{t:pi/2-a/2}],
['p342','p345',{a:pi-2*t}],
['p342','p343',{a:2*t}],
['p342','p343',{t:a/2}],
['p342','p331',{t:pi/2}],

['p343','p346',{t:pi/2-a/2}],
['p343','p346',{a:pi-2*t}],
['p343','p342',{a:2*t}],
['p343','p342',{t:a/2}],
['p343','p243',{a:pi}],


['p344','p345',{t:a}],
['p344','p345',{a:t}],
['p344','p341',{t:pi/2-a/2}],
['p344','p341',{a:pi-2*t}],

['p345','p344',{a:t}],
['p345','p344',{t:a}],
['p345','p346',{a:2*t}],
['p345','p346',{t:a/2}],
['p345','p342',{a:pi-2*t}],
['p345','p342',{t:pi/2-a/2}],

['p346','p345',{a:2*t}],
['p346','p345',{t:a/2}],
['p346','p343',{a:pi-2*t}],
['p346','p343',{t:pi/2-a/2}]
]


# List of regions that cover a=0. Should equal 0 when a=0.
zeroRegions = ['p346', 'p345', 'p344', 'p341', 'p332', 'p333', 'p323', 'p322', 'p321', 'p311']

# Run through all the comparisons. Need simplify(). Even together() gives some false negatives.

checkFile = open('/home/tim/Dropbox/PhD/Analysis/REM-chapter/checksFile.tex','w')

checkFile.write('All checks evaluated.\nTim Lucas - ' + str(datetime.now()) + '\n')
for i in range(len(allComps)):
        if (eval(allComps[i][0]).subs(allComps[i][2]) - eval(allComps[i][1]).subs(allComps[i][2])).simplify() == 0:
                checkFile.write(str(i) + ': ' + allComps[i][0]+ ' and ' +allComps[i][1]+': OK\n')
        else:
                checkFile.write(str(i) + ': ' + allComps[i][0]+ ' and ' +allComps[i][1]+': Incorrect\n')

for i in range(len(zeroRegions)):
        if eval(zeroRegions[i]).subs({a:0}).simplify() == 0:
                checkFile.write(zeroRegions[i] + ' at a=0: OK\n')
        else:
                checkFile.write(zeroRegions[i] + ' at a=0: Incorrect\n')

checkFile.close()


# And print to terminal
#for i in range(len(allComps)):
#        if not (eval(allComps[i][0]).subs(allComps[i][2]) - eval(allComps[i][1]).subs(allComps[i][2])).simplify() == 0:
#                print allComps[i][0] + ' and ' + allComps[i][1]+': Incorrect\n'

####################################
## Check some that don't work well ##
####################################

xRange = np.arange(0,pi/2, 0.01)
y332Range = [p332.subs({r:1,  t:pi/2, a:i}).n() for i in xRange]
plot332 = pl.plot(xRange, y332Range)
pl.savefig('/home/tim/Dropbox/PhD/Analysis/REM-chapter/imgs/p332Profile.pdf')
pl.close()

y341Range = [p341.subs({r:1,  t:pi/2, a:i}).n() for i in xRange]
plot341 = pl.plot(xRange, y341Range)
pl.savefig('/home/tim/Dropbox/PhD/Analysis/REM-chapter/imgs/p341Profile.pdf')
pl.close()



#pl.savefig('/home/tim/Dropbox/PhD/Analysis/REM-chapter/imgs/p221Profile.pdf')
#pl.close()
```

```


######################################################################
### Define a a function that calculates your answer.            ####
######################################################################

def calcP(A, T, R):
  assert (A <= 2*pi and A >= 0), "a is out of bounds. Should be in 0<a<2*pi"
  assert (T <= 2*pi and T >= 0), "s is out of bounds. Should be in 0<s<2*pi"

  if A > pi:
    if A < 4*pi - 2* T:
      p = p243.subs({a:A,  t:T, r:R}).n()
    elif A <= 3*pi -  T:
                        p = p222.subs({a:A,  t:T, r:R}).n()
    else:
                        p = p221.subs({a:A,  t:T, r:R}).n()
  else:
    if A < 4*pi - 2* T:
                        p = p322.subs({a:A,  t:T, r:R}).n()
    else:
                        p = p321.subs({a:A,  t:T, r:R}).n()
        return p


##############################
## Apply to entire grid   ###
##############################

# How many values for each parameter
nParas = 100

# Make a vector for a and s. Make an empty nParas x nParas array.
# Calculated profile sizes will go in pArray
tVec = np.linspace(0, 2*pi, nParas)
aVec = np.linspace(0, 2*pi, nParas)
pArray = np.zeros((nParas,nParas))

# Calculate profile size for each combination of parameters
for i in range(nParas):
        for j in range(nParas):
                pArray[i][j] = calcP(aVec[i], tVec[j], 1)

# Turn the array upside down so origin is at bottom left.
pImage = np.flipud(pArray)

# Plot and save.
pl.imshow(pImage, interpolation='none', cmap=pl.get_cmap('Blues') )
#pl.show()

pl.savefig('/home/tim/Dropbox/PhD/Analysis/REM-chapter/imgs/profilesCalculated.png')



#############################
#### Output R function.  ###
#############################

# To reduce mistakes, output R function directly from python.
# However, the if statements are not automatic.

Rfunc = open('/home/tim/Dropbox/PhD/Analysis/REM-chapter/calculateProfileWidth.R', 'w')

Rfunc.write("""calcProfileWidth <- function(alpha, theta, r){
        if(alpha > 2*pi | alpha < 0)
    stop('alpha is out of bounds. alpha should be in 0<a<2*pi')
        if(theta > 2*pi | theta < 0)
    stop('theta is out of bounds. theta should be in 0<a<2*pi')

  if(alpha > pi){
          if(alpha < 4*pi - 2*theta){
""" +
'           p <- ' + str(p243) +
'\n                } else if(alpha <= 3*pi - theta){'
'\n                        p <- ' + str(p222) +
'\n                } else {'
'\n                        p <- ' + str(p221) +
'\n                }'
'\n        } else {'
'\n         if(alpha < 4*pi - 2*theta){'
'\n                        p <- ' + str(p322) +
'\n     } else {'
'\n                        p <- ' + str(p321) +
'\n                }'
'\n        }'
'\n        return(p)'
'\n}'
```

```


)

Rfunc.close()
```

REM–Analysis.py

3. Supplementary Script: R implementation of models

This is a simple implementation of the models derived in the paper in R R Development Core Team (2010). Once given the parameters θ and α it automatically selects the correct model to apply.

```
calcProfileWidth <- function(theta_a, theta_s, r){
        if(theta_a > 2*pi | theta_a < 0)
    stop('theta_a is out of bounds. theta_a should be in 0<a<2*pi')
        if(theta_s > 2*pi | theta_s < 0)
    stop('theta_s is out of bounds. theta_s should be in 0<a<2*pi')

  if(theta_a > pi){
          if(theta_a < 4*pi - 2*theta_s){
            p <- r*(theta_s - cos(theta_a/2) + 1)/pi
                } else if(theta_a <= 3*pi - theta_s){
                        p <- r*(theta_s - cos(theta_a/2) + cos(theta_a/2 + theta_s))/pi
                } else {
                        p <- r*(theta_s + 2*sin(theta_s/2))/pi
                }
        } else {
          if(theta_a < 4*pi - 2*theta_s){
                        p <- r*(theta_s*sin(theta_a/2) - cos(theta_a/2) + 1)/pi
    } else {
                        p <- r*(theta_s*sin(theta_a/2) - cos(theta_a/2) + cos(theta_a/2 + theta_s))/pi
                }
        }
        return(p)
}
```

supplementaryRscript.R

References

R Development Core Team (2010) *R: A Language And Environment For Statistical Computing*. R Foundation For Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. 24

Rowcliffe, J., Field, J., Turvey, S. & Carbone, C. (2008) Estimating animal density using camera traps without the need for individual recognition. *Journal of Applied Ecology*, **45**, 1228–1236. 5, 9

SymPy Development Team (2014) *SymPy: Python library for symbolic mathematics*. 13