

CHAPTER 16

The ‘robust design’

William Kendall, USGS Colorado Cooperative Fish & Wildlife Research Unit

Changes in population size through time are a function of births, deaths, immigration, and emigration. Population biologists have devoted a disproportionate amount of time to models that assume immigration and emigration are non-existent (or, not important). However, modern thinking suggests that these effects are potentially (perhaps generally) quite important. For example, metapopulation dynamics are not possible without immigration and emigration in the subpopulations. A model which allows the estimation of emigration and immigration to a population is therefore of considerable utility.

In this chapter, we consider Pollock’s *robust design*, an approach which will allow us considerable flexibility in estimating a very large number of important demographic parameters, including estimates of emigration and immigration. As you might imagine, such a model is bound to be more complicated than most (if not all) of the models we’ve previously considered, but it brings more biological reality to the analysis of population dynamics.

16.1. Decomposing the probability of subsequent encounter

We begin by considering the probabilistic pathway that links two events – the initial capture, marking and live release of an individual, and its subsequent re-encounter (for the moment, we’ll focus on live encounters). We know by now that we can represent such an individual with the encounter history ‘11’. An individual that we mark and release but do not encounter on the subsequent sampling occasion would have the encounter history ‘10’. Back in Chapter 1, we motivated the need for estimating encounter probability by considering the utility of measures of *return rate*. You might recall that ‘return rate’* is *not* a robust measure of survival. Why? Well, recall from Chapter 1 that ‘return rate’ is, at minimum, the product of two events: (1) the probability of surviving from the time of initial mark and release to some future sampling occasion, and (2) the probability that the individual is encountered on that sampling occasion, conditional on being alive. Because the ‘return rate’ is in fact the product of two different probabilities, this makes it difficult (and frequently impossible) to determine if differences in ‘return rate’ are due to differences in the probability of survival, the probability of encounter, or both. To solve this problem, we introduced models which explicitly account for encounter probability, such that potential differences in survival probabilities can be determined.

* Recall the ‘return rate’ is simply the proportion of individuals marked and released at some occasion that are encountered on a subsequent occasion; in other words, ‘return rate’ is simply $x(11)/[x(11)+x(10)]$, where $x(11)$ is the number of individuals marked and encountered on a subsequent occasion, and $x(10)$ is the number marked and not encountered on a subsequent occasion.

In fact, our treatment of ‘return rate’ (both in the preceding paragraph, and in Chapter 1) is incomplete. It is incomplete because in fact ‘return rate’ is the product of more than two parameters – it is the product of at least 4 lower-level parameters. We can illustrate this dependence graphically, using a ‘fate diagram’, as indicated in Fig. (16.1):

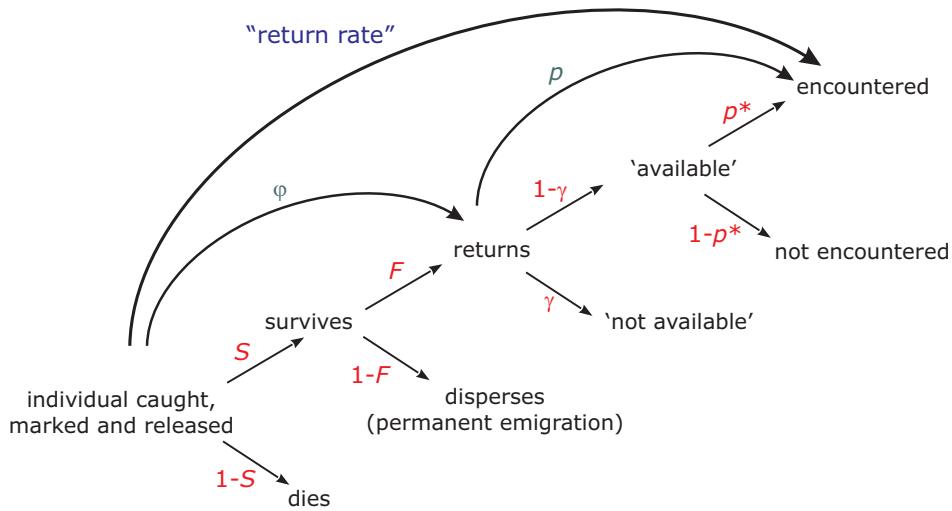


Figure 16.1: Basic fate diagram indicating the decomposition of ‘return rate’ into component transition parameters:

S (probability of surviving from release occasion i to subsequent sampling period $i+1$), F (probability that, conditional on surviving, that individual does not permanently leave (e.g., by permanent emigration) the population being sampled (i.e., the super-population; see Kendall 1999)), $(1-\gamma)$ (the probability that conditional on being alive, and in the super-population, that the individual is available to be encountered), and p^* (the probability that an individual is encountered, conditional on being alive, in the super-population, and available for encounter). The arcs indicate the underlying structure of apparent survival probability ($\varphi = S \times F$), apparent encounter probability ($p = (1 - \gamma) \times p^*$), and ‘return rate’ ($= S \times F \times (1 - \gamma) \times p^*$).

Starting at the lower left-hand corner of Fig. (16.1), we see that an individual animal is caught, marked and released alive at occasion (i). Then, there are several ‘events’ which determine if the individual is encountered alive at a subsequent sampling occasion ($i + 1$). First, the animal must survive – we use the parameter S to denote survival. The probability of the animal not surviving is given by the complement probability, $(1 - S)$. This much should be pretty obvious.

Next, conditional on surviving, a marked individual is potentially available for subsequent encounter if it remains in the ‘super-population’ (the larger population from which we are sampling). We use the parameter F to indicate the probability of fidelity of the marked individual to the super-population. We note that the fidelity parameter F was first introduced in Chapter 9, in the context of joint live encounter-dead recovery analysis. The complement, $(1 - F)$, is the probability that the animal has *permanently* left the super-population, e.g., by dispersing, and would thus not be available for subsequent *live* encounter in a sample drawn from this super-population under any circumstances.

Next, conditional on remaining in the super-population (with probability F), we introduce the concept of ‘availability’. It’s perhaps easiest to introduce this idea based on a simple biological example. Suppose we’re dealing with a bird species, where only breeding individuals are found at the breeding site where we conduct our encounter sampling. Clearly, then, only breeding individuals are ‘available’ for encounter, whereas non-breeding individuals would be ‘unavailable’. We model the probability of an individual being *unavailable* using the parameter γ (such that the probability of being *available* is given by its complement $1 - \gamma$).

Note that in most instances, the availability of a marked individual for encounter is conditional, varying from occasion to occasion (e.g., in some years, a marked individual breeds, and is thus available, whereas in other years, the same individual does not breed, and is thus unavailable). As such, we generally refer to the parameter γ as defining the probability that the marked individual has or has not *temporarily emigrated* from the study area. So, γ might be considered as the probability that the marked individual has temporarily emigrated from the study area. In fact, we'll see shortly that the γ parameter can be interpreted in more than one way.

Finally, conditional on surviving, remaining in the super-population, and being available for encounter, the marked individual is encountered live with probability p^* . Here, we use the asterisk '*' to differentiate what we will refer to as the 'true' encounter probability (p^*) from the 'apparent' encounter probability (p). The use of the familiar p to indicate apparent encounter probability is intentional, since it forces us to acknowledge that the familiar p parameter estimated in most models focused on live encounter data is in fact a 'function' of the true encounter rate, but is not true encounter rate in and of itself (except under very specific circumstances).

To make this clear, let's write out the following expression for 'return rate'. As noted earlier (and in Chapter 1), 'return rate' is in fact the product of two separate events – survival and encounter. But, we also noted that this simple definition is incomplete. It's incomplete, because it is more strictly correct to say that 'return rate' is the product of the *apparent* survival probability and the *apparent* encounter probability. If we let R represent return rate, and use φ and p to represent apparent survival rate and encounter probability, respectively, then we can write

$$R = (\varphi \times p).$$

Now, considering Fig. (16.1), we see that apparent survival (φ) is itself a product of true survival (S), and fidelity (F). This should make sense – the probability that an animal marked and released alive at occasion i will be encountered alive in the study area at occasion $(i+1)$ requires that the animal survives (with probability S), and remains in the super-population (with probability F ; if it permanently emigrates, then it will appear 'dead', since permanent emigration and mortality are confounded). So, $\varphi = SF$. Similarly, apparent encounter probability p is the product of the probability that the animal is available for encounter (with probability $1 - \gamma$), and the true detection probability p^* (which is the probability of detection, given availability, or presence). So, $p = (1 - \gamma)p^*$. Thus, we write

$$\begin{aligned} R &= \text{'apparent survival probability'} \times \text{'apparent encounter probability'} \\ &= (\varphi \times p) \\ &= (SF) \times ((1 - \gamma)p^*). \end{aligned}$$

Now, in several previous chapters, we simply decomposed 'return rate' R into apparent survival φ and apparent encounter probability p . The challenge, then, is to further decompose φ and p into their component pieces. In Chapter 9, we considered use of combined live encounter-dead recovery data to decompose φ . Recall that dead recovery data provides an estimate of true survival rate S , whereas live encounter data yields estimates of apparent survival probability φ . Since $\varphi = (SF)$, then an *ad hoc* estimate of F is given as $\hat{F} = (\varphi/S)$. The formal likelihood-based estimation of \hat{F} (described by Burnham, 1993) is covered in detail in Chapter 9.

What about the decomposition of apparent encounter probability p ? We see from Fig. (16.1) that $p = (1 - \gamma)p^*$. Following the logic we followed in the preceding paragraph to derive an *ad hoc* estimator for F , we see that $\hat{p}^* = \hat{p}/(1 - \hat{\gamma})$, and $\hat{\gamma} = 1 - (\hat{p}/\hat{p}^*)$; estimates of both the true encounter probability, and the 'availability' probability may be of significant interest.

16.2. Estimating γ : the classical 'live encounter' RD

The problem, then, is how to derive an estimate of either p^* or γ ? Recall that we can generate an estimate for p (apparent encounter probability) using our standard live encounter CJS models. But, where do we get estimates of γ , and p^* ? Are any of them estimated by any estimation model we've considered so far?

Well, if you think back to Chapter 15 (on closed population estimators), you might recall that one of the parameters estimated is in fact p^* . Now, in Chapter 15, we didn't refer to the parameter using the p^* notation, but with a few moments of thought, you should see they are essentially the same thing (well, not quite – recall that closed capture models estimate two different 'types' of encounter probability – p and c – we'll deal with these details later). In a closed population, there is neither entry or exit of individuals (i.e., N is a constant). As such, your estimate of the encounter probability is not conditional on presence or availability, since (by definition for a closed population) the marked individuals must be there. So, the estimate of p from a closed population model allows you to derive an estimate of p^* (given $n > 1$ occasions, $p^* = 1 - [(1 - p_1)(1 - p_2) \dots (1 - p_n)]$).

OK, fine, but why is this important? It's important because *if* you have an estimate of apparent encounter probability p , and *if* you have an estimate of true encounter probability p^* , then you can derive an *ad hoc* estimate of γ as $\hat{\gamma} = 1 - (\hat{p}/\hat{p}^*)$.

Now, for the 'big leap forward'. To derive the estimate of γ , we need an estimate of p (which we can get from standard open, live encounter CJS models), and p^* (which we can get from standard closed estimates). Can we derive both estimates from the same data set (based on samples from the same population)?

The answer (as first described by Ken Pollock) is 'yes' – by application of what has been described as the *robust design*. The robust design model is a *combination* of the Cormack-Jolly-Seber (CJS) (Cormack 1964, Jolly 1965, Seber 1965) live recapture model and the closed capture models. The model is described in detail by Kendall *et al.* (1997, 1995) and Kendall & Nichols (1995), and is represented schematically in standard ('classical') form in Fig. (16.2):

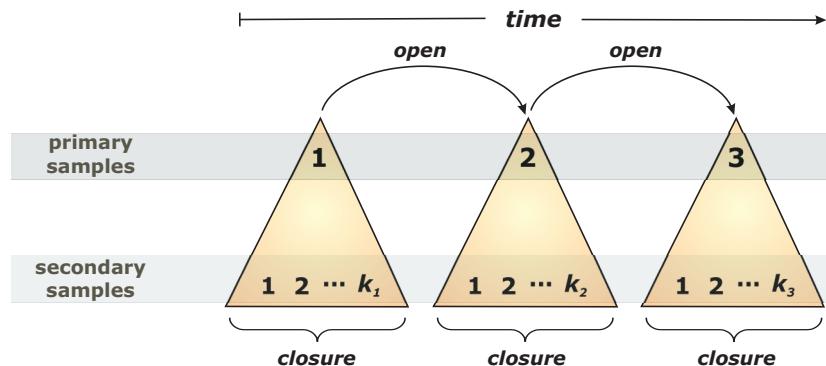


Figure 16.2: Sampling structure of the 'classical' Pollock's robust design.

The key difference from the standard CJS model considered in several earlier chapters is that instead of just one capture occasion between survival intervals, multiple (>1) capture occasions are used. These occasions are close together in time – so close that it allows you to (in general) 'safely' assume that the populations are closed while these samples are being taken (i.e., no mortality or emigration occurs during these short time intervals).

In fact, Pollock pointed out that in many cases data were being collected in this way anyway (e.g., small mammal sampling might be conducted in groups of 5-7 consecutive trapping days). The closed encounter occasions are termed *secondary* trapping occasions, and each primary trapping session can be viewed as a closed capture survey.

The power of this model is derived from the fact that, in addition to providing estimates of abundance (\hat{N}), the probability that an animal is captured at least once in a trapping session can be estimated from the data collected during the session using capture-recapture models developed for closed populations (Chapter 15)*. The longer intervals between *primary* trapping sessions allows estimation of survival, temporary emigration from the trapping area, and immigration of marked animals back to the trapping area. If the interval between primary sampling sessions is sufficiently long, gains (birth and immigration) and losses (death and emigration) to the population can occur. This contrasts with secondary samples (within the primary sampling session), where the interval between samples is sufficiently short that the population is effectively closed to gains and losses.

Recall that we're seeking estimates of both p and p^* , from which we can derive an estimate of γ . The relationship of the various parameters to the standard robust design is shown in Fig. (16.3):

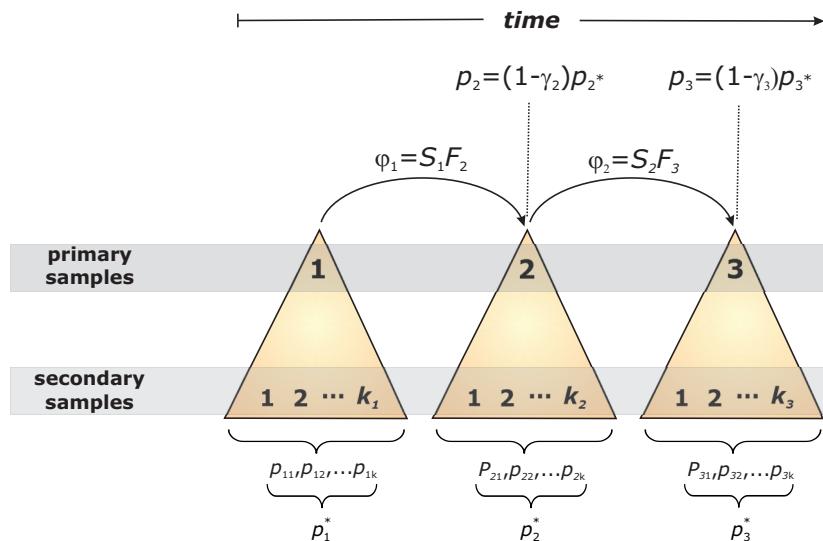


Figure 16.3: Relationship of key parameters to basic sampling structure of Pollock's robust design.

For each secondary trapping session (i), the probability of first capture p_{ij} and the probability of recapture c_{ij} are estimated (where j indexes the number of trapping occasions within the session), along with the number of animals in the population that are on the trapping area N_i . For the intervals between trapping sessions (i.e., between primary sessions, when the population is open), the probability of apparent survival φ_i ($= S \times F$), and the apparent encounter probability p are estimated.

It is clear from Fig. (16.3) that it should be possible to derive estimates of γ . In the absence of extra information (specifically, dead recovery data, or the equivalent), partitioning apparent survival φ into component elements S and F is not feasible using the classical robust design (which is based entirely on live encounters at a single location). We will deal with extensions to the classic robust design later in this chapter.

* In fact, we can use the structural framework of the robust design to estimate abundance (under some assumptions) even for situations where there is only a single sampling event per primary sample (e.g., CJS open population models) – see the Addendum to this chapter.

16.3. The RD extended – temporary emigration: γ' and γ''

Earlier we introduced the parameter γ as the probability that the individual was ‘unavailable’ for encounter at some particular primary sampling session. Kendall *et al.* (1995a, 1997) extended the simple (classical) parameterization of the robust design in terms of parameter γ by introducing two different parameters: γ' and γ'' (read as ‘gamma-prime’ and ‘gamma-double-prime’, respectively). These two new parameters are defined as follows:

parameter	definition
γ'_i	the probability of being <i>off</i> the study area, unavailable for capture during primary trapping session (i) given that the animal <i>was not</i> present on the study area during primary trapping session ($i - 1$), and survives to trapping session (i).
γ''_i	the probability of being <i>off</i> the study area, unavailable for capture during the primary trapping session (i) given that the animal <i>was</i> present during primary trapping session ($i - 1$), and survives to trapping session (i).

Now, these are perhaps more difficult to ‘wrap your brain around’ than they might first appear. You need to read the definitions carefully.

First, we distinguish between the ‘observable’ (i.e., potentially available for encounter at time (i)) and ‘unobservable’ (i.e., potentially unavailable for encounter at time (i)) parts of the population of interest (Fig. 16.4). The ‘superpopulation’ (i.e., the target population of interest) is the sum of the ‘observable’ and ‘unobservable’ individuals.

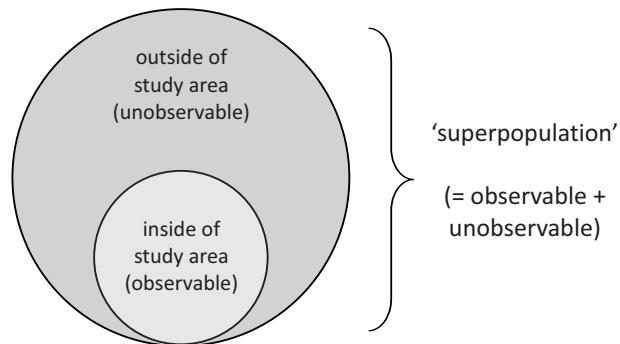
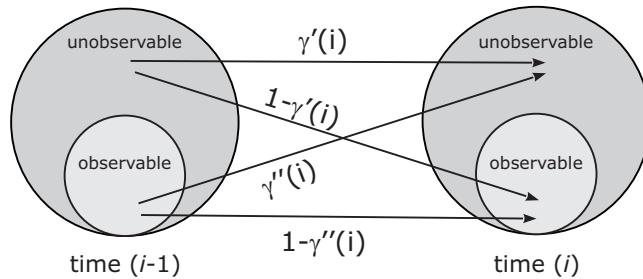


Figure 16.4: Relationships between observable (i.e., available to be encountered during sampling), and unobservable (i.e., not available to be encountered during sampling) segments of the population of interest. The larger circle represents the range of the super-population. The smaller circle (light grey) represents the part of the superpopulation that is available for encounter (i.e., in the study area), whereas the darker part of the larger circle represents individuals unavailable for encounter (i.e., temporarily outside the study area).

The γ parameters introduced by Kendall define the probability of movement between the ‘observable’ and ‘unobservable’ states, between any two time steps. The basic relationship between γ' and γ'' is shown in Fig. (16.5). Start with the parameter γ''_i . It is the probability that given that you were available at time ($i - 1$), that you are not available now at time (i). In other words, γ'' is the probability of an individual that is available for encounter at time ($i - 1$) temporarily emigrating between time ($i - 1$) and (i), such that it is not available for encounter at time (i). Thus, $(1 - \gamma'')$ is the probability of being in the

Figure 16.5: Relationships between γ' and γ'' .

study area at time (i), given that it was also in the sample at time ($i - 1$). In other words, ‘was in last occasion, stayed in this occasion’.

As indicated in Fig. (16.5), the parameter γ''_i is the probability of temporarily emigrating from the sample between sampling occasions ($i - 1$) and (i), and its complement ($1 - \gamma''_i$) is the probability of remaining in the sample between sampling occasions ($i - 1$) and (i).

What about parameter γ' ? Again, consider Fig. (16.5) – γ' is the probability that given that an individual was *not* in the sample at time ($i - 1$), is also *not* present (i.e., not in the sample) at time (i). In effect, γ' is the probability of remaining outside the sample (‘was out last occasion, stayed out this occasion’). Thus, $(1 - \gamma')$ is the probability that an individual which *was* out of the sample at time ($i - 1$) *enters* the sample between time ($i - 1$) and time (i) – i.e., *return rate of temporary emigrants*.

To keep track of the different γ parameters, and to reinforce the fact that the γ parameters relate to temporary movements into or out of the observable sample, we can consider the γ parameters as the probabilities of a *state transition matrix*, mapping state (observable, unobservable) now (time (i)), and state at the next time step (time ($i + 1$)) :

	unobservable, time (i)	observable, time (i)
unobservable, ($i + 1$)	γ'_i	γ''_i
observable, ($i + 1$)	$1 - \gamma'_i$	$1 - \gamma''_i$

Indexing of these parameters (as indicated in Fig. 16.5) follows the notation of Kendall *et al.* (1997). Thus, γ''_2 applies to the interval before the second primary trapping session. It is important to note that not all parameters are estimable (either because of *logical constraints*, or *statistical confounding*).

For example, parameter γ'_2 is not estimated because there are no marked animals outside the study area at primary trapping session 2 that were also outside the study area at time 1 (because they could not have been marked otherwise). In general, for a study with k primary sessions, (i) S_1, S_2, \dots, S_{k-1} , (ii) $p_{ij}, i = 1 \dots k, j = 1 \dots k_i$, (iii) $\gamma'_3, \gamma'_4, \dots, \gamma'_{k-1}$, and (iv) $\gamma''_2, \gamma''_3, \dots, \gamma''_{k-1}$ are estimable. General issues of estimability of various parameters is discussed elsewhere (below).

16.3.1. γ parameters and multi-state notation

If these parameters are still confusing, note the similarity of Fig. (16.5) to multi-state models introduced in Chapter 10. In fact, this temporary emigration model is a special case of a multi-state model with two states. Defining state **O** to be the study area (**O**; observable) and state **U** to be off the study area (**U**;

unobservable), then $\gamma'' = \psi^{OU}$ and $\gamma' = \psi^{UU}$. The basic relationship between the ψ parameters and the ‘observable’ and ‘unobservable’ states is shown in Fig. (16.6):

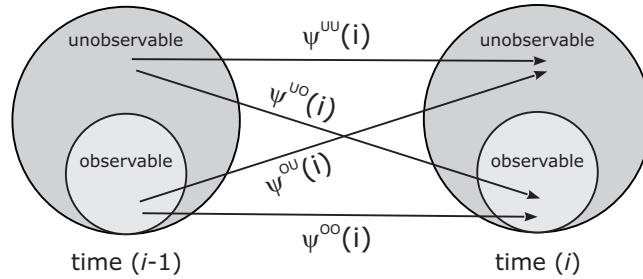


Figure 16.6: Multi-state (ψ) probabilities of transition between ‘observable’ and ‘unobservable’ states.

If you compare Fig. (16.6) with Fig. (16.5) for a few moments, you should recognize that

$$\begin{aligned}\gamma' &\equiv \psi^{UU} \\ 1 - \gamma' &\equiv \psi^{UO} \\ \gamma'' &\equiv \psi^{OU} \\ 1 - \gamma'' &\equiv \psi^{OO}\end{aligned}$$

In fact, you could, with a bit of work, perform a ‘typical’ single sampling location robust design problem as a multi-state problem with two states – you would simply fix $S = 1$ and $\psi^{OU} = 0$ in the closed periods (modeling the encounter probability only). In the ‘**Closed Robust Design Multi-state**’ and ‘**Open Robust Design Multi-state**’ options in MARK, which we describe later in this chapter, we abandon the use of the ‘ γ notation’ altogether. Although those models are more flexible, the models using γ that we are discussing here are much simpler to set up.

16.3.2. illustrating the extended model: encounter histories & probability expressions

To illustrate the mechanics of fitting the classical robust design model, assume a simple case with 3 primary trapping sessions, each consisting of 3 secondary trapping occasions. The encounter history in its entirety is viewed as 9 live capture occasions, but with unequal spacing. Thus, the encounter history might be viewed as

1 1 1 → 1 1 1 → 1 1 1

where the ‘ \rightarrow ’ separates the primary trapping sessions. The probability that an animal is captured at least once during a trapping session is defined as p_i^* (see Chapter 15), and is estimated as

$$p_i^* = 1 - [(1 - p_{i1}) \times (1 - p_{i2}) \times (1 - p_{i3})].$$

That is, the probability of not seeing an animal on trapping occasion j is $(1 - p_{ij})$ for $j = 1, 2$, and 3.

The probability of *never* seeing the animal during trapping session i is

$$(1 - p_{i1}) \times (1 - p_{i2}) \times (1 - p_{i3}),$$

so therefore, the probability of seeing the animal at least once during the trapping session is 1 minus this quantity. Note that the p_{ij} are estimated as with the closed capture models (Chapter 15).

To illustrate the meaning of the emigration (γ''_i) and immigration (γ'_i) parameters, suppose the animal is captured during the first trapping session, not captured during the second trapping session, and then captured during the third trapping session. One of many encounter histories that would demonstrate this scenario would be (where spaces in the encounter history separate primary sampling sessions, but which would not appear in an actual encounter history):

010 000 111

which, if pooled over secondary samples within primary samples, would be equivalent to the encounter history '101'.

The probability of observing this 'pooled' encounter history can be broken down into 2 parts. First, consider the portion of the probability associated with the *primary* intervals. This would be

$$\varphi_1 \varphi_2 \left[\gamma''_2 (1 - \gamma'_3) + (1 - \gamma''_2) (1 - p_2^*) (1 - \gamma''_3) \right] p_3^*.$$

The product in front of the first bracket [$\varphi_1 \varphi_2$] is the probability that the individual survived from the first primary trapping session to the third primary trapping session. Because we encountered it alive on the third occasion (i.e., at least once during the three secondary trapping sessions during the third primary session), we know the individual survived both intervals (this is a logical necessity, obviously).

The complicated-looking term in the brackets represents the probability that the individual was not captured during the second trapping session. The first product within the brackets [$\gamma''_2 (1 - \gamma'_3)$] is the probability that the individual emigrated between the first 2 primary trapping sessions (γ''_2), and then immigrated back onto the study area during the interval between the second and third trapping sessions [$1 - \gamma'_3$]. However, a second possibility exists for why the animal was not captured, i.e., that it remained on the study area and just was not captured. The term [$1 - \gamma''_2$] represents the probability that the individual 'remained on the study area'. The term [$1 - p_2^*$] represents individuals 'not captured'. The final term [$1 - \gamma''_3$] represents the probability that the individual remained on the study area so that it was available for capture during the third trapping session.

The second portion of the cell probability for the preceding encounter history (p_3^*) involves the estimates of p_i^* , and is thus just the closed capture model probabilities.

16.3.3. Random (classical) versus Markovian temporary emigration

The probability of movement between 'availability states' can be either *random*, or *Markovian*. If the former (random), the probability of moving between availability states between primary occasions (i) and ($i + 1$) is independent of the previous state of the system, whereas for Markovian movement, the probability of moving between availability states between primary occasions (i) and ($i + 1$) is conditional on the state of the individual at time ($i - 1$). Note that random movement is essentially what was assumed under the classical robust design model discussed earlier (i.e., the RD model based on γ , and not parameterized in terms of γ' and γ'').

To provide identifiability of the parameters for the '*Markovian emigration*' model (where an animal

'remembers' that it is off the study area) when parameters are time-specific, Kendall *et al.* (1997) stated that γ''_k and γ'_k need to be set equal to γ''_t and γ'_t , respectively, for some earlier period. Otherwise these parameters are confounded with S_{t-1} . They suggested setting them equal to γ''_{k-1} and γ'_{k-1} , respectively, but it really should depend on what makes the most sense for your situation. This confounding problem goes away if either movement or survival is modeled as constant over time.

To obtain the '*Random emigration*' model, set $\gamma'_i = \gamma''_i$. This constraint is perhaps not intuitively obvious. The interpretation is that the probability of temporarily emigrating from the observable sample during an interval is the same as the probability of staying away (i.e., the probability of not immigrating back into the observable sample). Biologically, the probability of being in the study area during the current trapping session is the same for those animals previously in and those animals previously out of the study area during the previous trapping session. The last survival parameter, S_{k-1} , is also not estimable under the time-dependent model unless constraints are imposed. That is, the parameters γ''_k , γ'_k , and S_{k-1} are all confounded. Setting the constraints $\gamma''_{k-1} = \gamma''_k$ and $\gamma'_{k-1} = \gamma'_k$, for example, makes the resulting 3 parameters estimable. Or, you could forgo the constraint – in that case, you would simply ignore the estimates of S_{k-1} , γ'_k , and γ''_k . Estimates of the remaining parameters would be unbiased.

The null model for both the random and Markovian models is the '*No emigration*' model. To obtain the '*No emigration*' model, you simply set all the γ parameters to zero. If all the γ''_i are set to zero, then the γ'_i must all be set to zero also, because there are no animals allowed to emigrate to provide a source of immigrants back into the population.

To make the distinction between the random (classical) and Markovian temporary emigration robust design models clearer, consider the cell probability expressions for the following encounter history:

110 000 010 111

Here, we have 4 primary trapping occasions, and 3 secondary trapping occasions per primary occasion. If we considered only primary occasions, the encounter history for this individual would be '1011'. The individual was marked and released on the first secondary occasion within the first primary sampling occasion, and then seen again on the second secondary occasion within that first primary period. The individual was not seen at all during any of the secondary samples during the second primary sampling occasion. The individual was seen once – on the second of the secondary sampling occasions – during the third primary sampling occasions, and was seen on all of the secondary sampling occasions during the final primary sampling period.

Again, what is key here is the second primary sampling occasion – during the second primary occasion, the individual was not seen at all. This might occur in one of three ways. First, the individual could have died – we assume only live encounters are possible. However, since the individual was seen alive at least once on a subsequent primary sample, then we clearly cannot assume that the '000' secondary encounter history on the second primary occasion reflects death of the individual.

However, there are two other possibilities we need to consider:

1. the individual could be alive and in the observable sample, but simply 'missed' (i.e., not encountered),

or, alternatively,

2. the individual could have temporarily emigrated from the observable sampling region between primary occasion 1 and primary occasion 2, such that it is unavailable for encounter during primary occasion 2 (i.e., is unobservable).

We have to account for both possibilities when constructing the probability statements. The following

table shows the probability expressions for both the Markovian and random temporary emigration models:

model	probability
Markovian	$\varphi_1\gamma_2''\varphi_2(1-\gamma_3')p_3^*\varphi_3(1-\gamma_4'')p_4^*$ + $\varphi_1(1-\gamma_2'')(1-p_2^*)\varphi_2(1-\gamma_3'')p_3^*\varphi_3(1-\gamma_4'')p_4^*$
random	$\varphi_1\gamma_2\varphi_2(1-\gamma_3)p_3^*\varphi_3(1-\gamma_4)p_4^*$ + $\varphi_1(1-\gamma_2)(1-p_2^*)\varphi_2(1-\gamma_3)p_3^*\varphi_3(1-\gamma_4)p_4^*$

Look at the tabulated probability expressions carefully. Make sure you understand the distinction between the random and Markovian temporary emigration models, and how the various constraints needed for identifiability affect the probability expressions.

Now, let’s step through each expression, to make sure you see how they were constructed. Let’s start with the Markovian emigration expression. Note that the probability expression for both models is written in two pieces (separated by the ‘+’ sign). These two pieces reflect the fact that we need to account for the two possible ways by which we could achieve the ‘000’ encounter history for the second primary sampling occasion: either (i) the individual was not available to be sampled (with probability γ_2'' ; in other words, it was in the sample at primary occasion 1, and left the sample at primary occasion 2, such that it was unavailable for encounter), or (ii) was in the sample during primary sampling occasion 2, with probability $(1 - \gamma_2'')$, but was simply missed (i.e., not encountered).

Let’s consider the first part of the probability expression. Clearly, φ_1 indicates the individual survived from primary occasion 1 → 2. We know this to be true. The γ_2'' term indicates the possibility that the individual temporarily emigrated from the sample between occasions 1 and 2, such that it was unavailable for encounter during primary sampling occasion 2. Then, φ_2 , since the individual clearly survives from occasion 2 to occasion 3. Then, conditional on having temporarily emigrated at occasion 2, we need to account for the re-entry (immigration) back into the sample at occasion 3, with probability $(1 - \gamma_3')$. This is logically necessary since the individual was encountered at least once during primary sampling occasion 3. Next, φ_3 , since the individual clearly survives from occasion 3 to 4. Finally, the individual stays in the sample (since it was encountered), with probability $(1 - \gamma_4'')$, and was encountered with probability p_4^* .

Now, the second term of the expression (after the ‘+’ sign) is similar, with one important difference – in the second term, we account for the possibility that the individual stayed in the sample between primary sampling occasion 1 and 2 with probability $(1 - \gamma_2'')$, and was not encountered during any of the secondary samples during primary sampling occasion 2 with probability $(1 - p_2^*)$.

For the random emigration model, the expressions are the same, except we’ve eliminated the ‘primes’ for the γ terms (we note that we could, with a bit of algebra, reduce both expressions to simpler forms – especially the expression for random emigration. However, leaving the expressions in ‘expanded’ form makes the logic of how the expressions were constructed more obvious).

16.3.4. Alternate movement models: no movement, and ‘even flow’

While in the preceding we focussed on contrasting random and Markovian movement models, it is clear that both need to be tested against an explicit null of ‘No movement’. For this null model, we assume that individuals that are ‘observable’ are always ‘observable’ over all sampling occasions.

Similarly, individuals which are ‘unobservable’ remain unobservable over all sampling occasions. We construct the ‘No movement’ fairly easily, by simply setting the γ ’s to 1 (unobservable individuals remain unobservable) and γ'' ’s to 0 (observable individuals remain observable).^{*} Unless you have compelling evidence to the contrary, it is always worth including a ‘No movement’ model in your candidate model set.

Another, somewhat more subtle model, is what we might call an ‘Even flow’ model. In the ‘Even flow’ model, we are interested in whether the probability of moving from ‘observable’ at time (i) to ‘unobservable’ at time ($i+1$) is the same as the probability of moving from ‘unobservable’ to ‘observable’ over the same time interval. In other words, $(1 - \gamma') = \gamma''$. Note that the ‘even flow’ model says only that the *per capita* probability of moving to the alternate state over some interval is independent of the originating state at the start of the interval.

Be sure you understand the distinction between the ‘Even flow’ model and the ‘Random movement’ and ‘Markovian movement’ models. In the ‘Random movement’ model we set $\gamma' = \gamma''$, which means that the probability of an individual being unobservable at time ($i+1$) is independent of whether or not it was ‘observable’ at time (i). As noted earlier, the interpretation is that the probability of *emigrating* during an interval is the same as the probability of staying away (conditional on already being ‘unobservable’ at the start of the interval). For the ‘Markovian movement’ model, we allow for movement rates to differ as a function of whether the individual is ‘observable’ or ‘unobservable’ – the only constraints we apply to the γ parameters in the Markovian model are necessary to ensure identifiability. Contrast this with the ‘Even flow’ model, where we enforce an equality constraint between entry and exit from a given state over the interval. We will leave it to you to decide which of these models are sufficiently ‘biologically plausible’ to consider including in your candidate model set.

The following table (16.1) summarizes some of the constraints which are commonly used to specify (and in some case, make identifiable) the 4 model types we’ve discussed so far (‘No movement’, ‘Random movement’, ‘Markovian movement’, and ‘Even flow’):

Table 16.1: Parameter constraints for standard model types using classical closed RD (γ) parameterization.

model	constraint
no movement	$\gamma' = 1, \gamma'' = 0$
random movement	$\gamma' = \gamma''$
Markovian movement	$\gamma'_k = \gamma'_{k-1}$ $\gamma''_k = \gamma''_{k-1}$
‘even flow’	$\gamma'' = (1 - \gamma')$

* Practically speaking, it would not matter if you fixed γ' to 1 or 0. Since the model does not consider movement of marked animals outside the study area, γ' never enters the likelihood and therefore it doesn’t matter whether you fix it to 0 or 1. However, setting $\gamma' = 1$ for a ‘no movement’ model is logically more consistent with Fig. (16.5).

16.4. Advantages of the RD

Advantages of the robust design alluded to above include:

1. estimates of p_i^* , and thus N_i and recruitment are less biased by heterogeneity in capture probability (specifically, if you use heterogeneity models within season; see Chapter 15)
2. temporary emigration can be estimated assuming (i) completely random, (ii) Markovian, or (iii) temporarily trap dependent availability for capture (Kendall & Nichols 1995, Kendall *et al.* 1997)
3. If temporary emigration does not occur, then abundance, survival, and recruitment can be estimated for all time periods (e.g., in a 4-period study, half the parameters are inestimable using the JS method; Kendall & Pollock 1992).
4. Precision tends to be better using the formal robust design models of Kendall *et al.* (1995), which include the model described above with $\gamma'' = \gamma' = 0$.
5. Because there is information on capture for the youngest catchable age class, estimation of recruitment into the second age class can be separated into *in situ* recruitment and immigration when there are only 2 identifiable age classes. Using the classic design (i.e., one capture session per period of interest), 3 identifiable age classes are required (Nichols & Pollock 1990).
6. The robust design's 2 levels of sampling allow for finer control over the relative precision of each parameter (Kendall & Pollock 1992).

16.5. Assumptions of analysis under the RD

For the most part, the assumptions under the robust design are a combination of the assumptions for closed-population methods and the JS method:

1. Under the classical robust design (as first described by Ken Pollock, and subsequently extended by Kendall and colleagues; hereafter, we refer to this as the *closed robust design*), the population is assumed closed to additions and deletions across all secondary sampling occasions within a primary sampling session. Kendall (1999) identified 3 scenarios where estimation of p_i^* would still be unbiased when closure was violated.
 - a. If movement in and out of the study area, or more generally, if the availability of individuals for detection is completely random, then \hat{p}_i^* remains unbiased but now a function of 'effective capture probability' (i.e., $P(\text{availability}) \times P(\text{detection} | \text{availability})$). So now \hat{N} estimates the size of the super-population, the individuals associated with the study area *for that primary period*, but not necessarily present at any given sampling occasion. The other 2 exceptions require that detection probability vary only by time and might apply most with migratory populations.
 - b. If the entire population is present at the first session within a period but begins to leave before the last session, then the estimator is unbiased if detection histories are pooled for all sessions that follow the first exit from the study area. If the exodus begins after the first session this creates a new 2-session detection history within period.
 - c. Conversely, if sampling begins before all animals in the population have arrived but they are all present in the last session, then all sessions up to the point of first entry should be pooled.

2. Temporary emigration is assumed to be either completely random, Markovian, or based on a temporary response to first capture.
3. Survival probability is assumed to be the same for all animals in the population, regardless of availability for capture. This is a strong assumption, especially in the Markovian availability case.

16.6. RD (closed) in MARK – some worked examples

OK, enough of the background for now. Let's actually use the closed robust design in **MARK**. We'll begin with a very simple example which can be addressed using only PIMs and the PIM chart, followed by a more complex model requiring modification(s) of the design matrix.

16.6.1. Closed robust design – simple worked example

We'll demonstrate the 'basics' using some data simulated under a '*Markovian movement*' model. The data (contained in **rd_simple1.inp**) consist of 3,000 individuals in a study area, some of which are captured, marked and released alive. Each of the 5 primary sampling sessions consisted of 3 secondary samples. So, in total, $(5 \times 3) = 15$ sampling occasions. For our simulation, we assumed that survival between primary periods varied over time: $S_1 = 0.7, S_2 = 0.8, S_3 = 0.9, S_4 = 0.8$. Within each year, we assumed that the true model for encounters during the secondary samples was model $\{p(\cdot) = c(\cdot)\}$ (i.e., model M_0 – see Chapter 15). We used $p_{11 \rightarrow 13} = 0.5, p_{21 \rightarrow 33} = 0.6, p_{41 \rightarrow 43} = 0.5$ and $p_{51 \rightarrow 53} = 0.5$. (*Note:* setting $p_{11} = p_{12} = p_{13} = 0.5$ implies that p_1^* , the probability of being captured at least once in primary period 1, is $p_1^* = 1 - [(1 - 0.5)(1 - 0.5)(1 - 0.5)] = 0.875$).

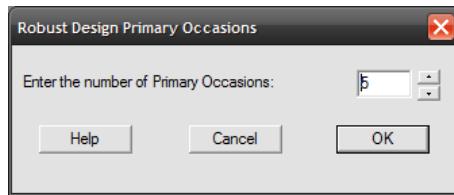
If you notice, the total number of individuals captured at least once in primary session 1 in the simulated data set is 2,619, which is close to the expected value of $3,000 \times 0.875 = 2,625$. We also assumed (purely for convenience) that no individual entered the population between the start and end of the study (thus, since $S < 1$, the estimated population size should decline over time). We also assumed no heterogeneity in capture probabilities among individuals. What about the γ parameters? We assumed a time-dependent Markovian model: $\gamma_2'' = 0.2, \gamma_3'' = 0.3, \gamma_4'' = 0.3, \gamma_5'' = 0.2$ and $\gamma_3' = 0.2, \gamma_4' = 0.4, \gamma_5' = 0.3$.

OK, now, let's analyze these simulated data in **MARK**. For our candidate model set, we'll assume that there are 3 competing models: (i) a model with no temporary emigration (i.e., $\gamma_i'' = \gamma_i' = 0$), (ii) a model with random temporary emigration (i.e., $\gamma_i'' = \gamma_i'$), and (iii) a model with Markovian temporary emigration (in this case, the 'true' model under which the data were simulated). We'll skip the 'even flow' model mentioned earlier for now. It is not a model we can build directly using PIMs. Moreover, building the 'even flow' DM requires a design matrix 'trick' we haven't seen before. For now, we're going to concentrate on simple model construction, using PIMs. In our analysis, we'll also assume we have 'prior knowledge' concerning the true structure for the encounter probabilities (i.e., the parameter structure for p_i and c_i). To facilitate referring to the models in the results browser, we'll call them simply 'No movement', 'Random movement' and 'Markovian movement', respectively.

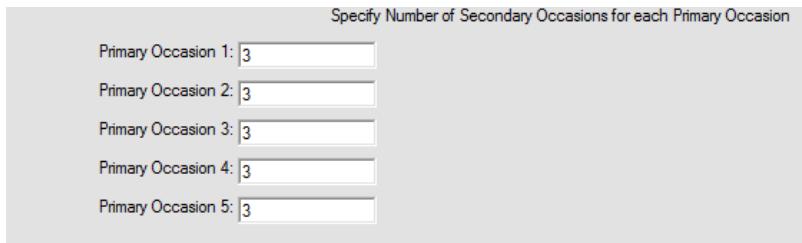
Start **MARK** and select the '**Robust Design**' data type on the model specification window. **MARK** will immediately 'pop-up' a small sub-window, asking you specify the model type for the closed captures data type (recall that you're modeling encounters during secondary samples using a closed population estimator). For this example, we'll use '**Huggins p and c**'. After selecting the appropriate input file (**rd_simple1.inp**), we need to tell **MARK** how many occasions we have. For the robust design, we need to do this in stages. First, how many total occasions? In this case, we have 5 primary occasions, each of which consists of 3 secondary occasions. So, 15 total occasions.

The next stage is specifying the primary and secondary sampling structure. In other words, how are the secondary samples divided among primary samples? If you look at the .INP file, there is no obvious indication in the file itself where the break-points are between primary occasions. However, MARK has a useful feature which makes specifying the primary and secondary sample structure relatively straightforward. If you look immediately to the right of where you entered the total number of occasions, you'll see the usual '**Set Time Intervals**' button. Immediately above and to the right of the '**Set Time Intervals**' button is a button labeled '**Easy Robust Design Times**'. Why 2 buttons? Well, you could specify the primary and secondary sampling model structure by appropriately setting the time intervals (see below), or you can take the 'easy way out' (pun intended) by using the '**Easy Robust Design Times**' button.

If you click this button, you're presented with a new window which asks you to specify the number of primary sampling occasions:



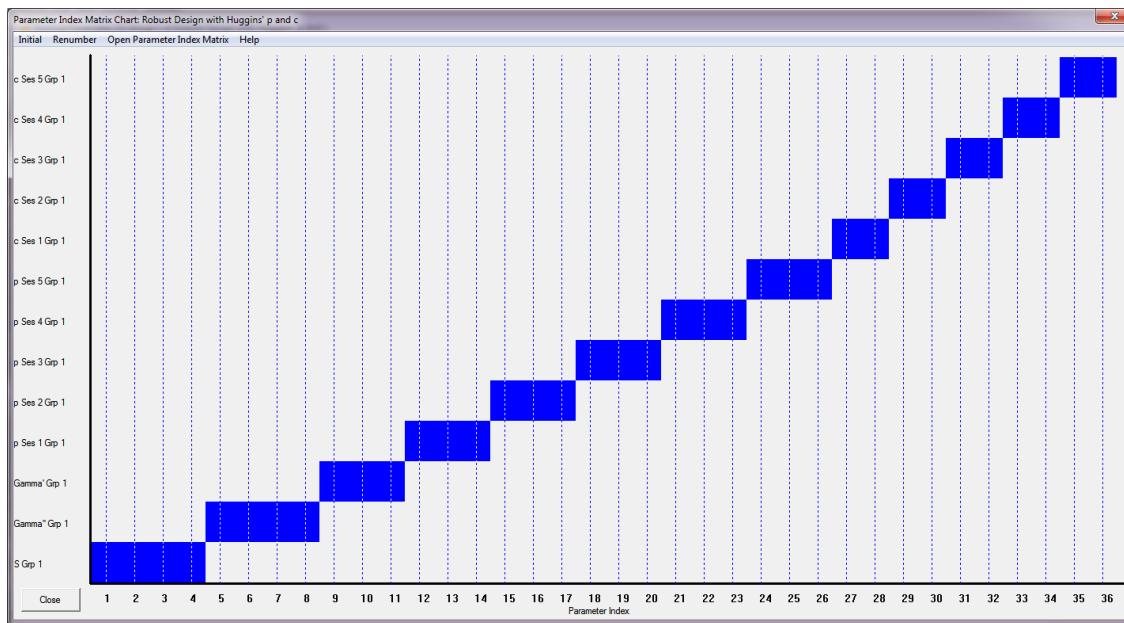
In our case, we have 5 primary sampling occasions. Once you click the '**OK**' button, MARK responds with a second pop-up window, asking you to specify the number of secondary sampling occasions for each primary session.



The default values that you will see are derived simply by taking the total number of occasions (15 in this example) and dividing that number by the number of primary sampling occasions (5) – in this example, the default of 3 secondary sampling occasions conveniently matches the true structure of our sampling – of course, if it didn't, then we would simply manually adjust the number of secondary sampling occasions per primary sampling occasion, subject to the constraint that the total number of secondary occasions (summed over all primary sampling occasions) equaled 15 (in our example).

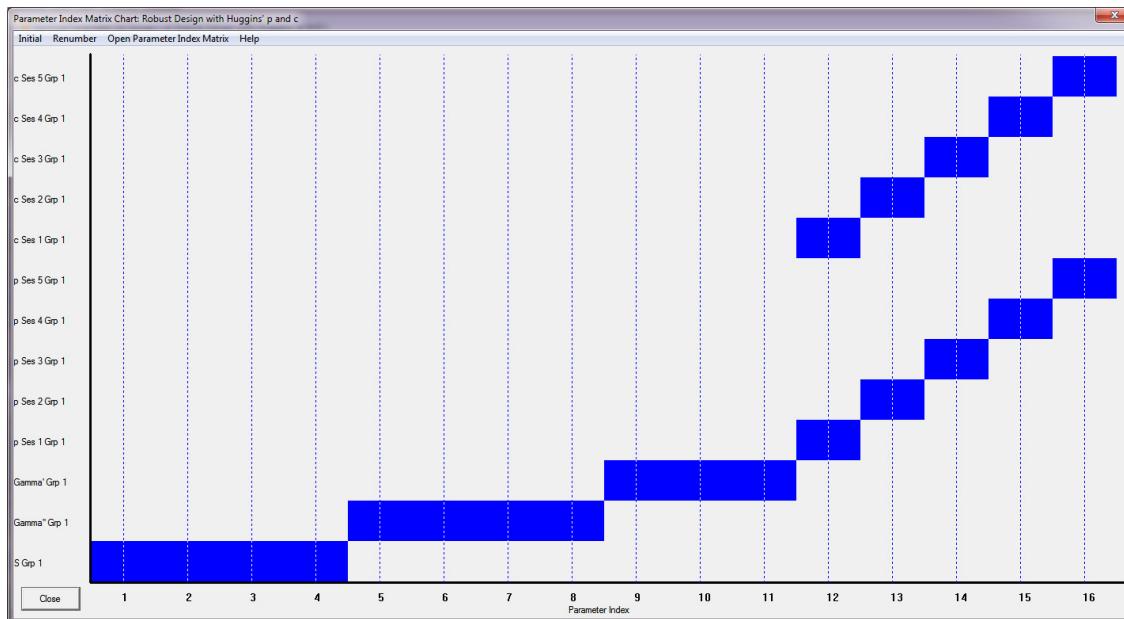
Once you have correctly specified the primary and secondary sampling structure (by whichever method you chose), click the '**OK**' button. As usual, MARK responds by presenting you with the PIM for the first parameter, in this case, the survival parameter S .

Let's look at the PIM chart – but, remember how many parameters you're dealing with here: you have survival (S), the γ parameters (γ' and γ''), and the two encounter parameters (p and c). Meaning, the PIM chart will be quite big. Even for this simple example, with 'only' 15 total occasions, the PIM chart (shown at the top of the next page) is 'dense' with information (to put it mildly):



We'll start by trying to fit a model with time dependence in S , γ' , and γ'' , but where $p_i = c_i = c$ for each primary occasion (i) (although we allow annual p to vary).

The PIM chart corresponding to this model is shown below (notice how much 'smaller' and 'less dense' this PIM chart is, reflecting the reduction in the number of parameters from $36 \rightarrow 16$):



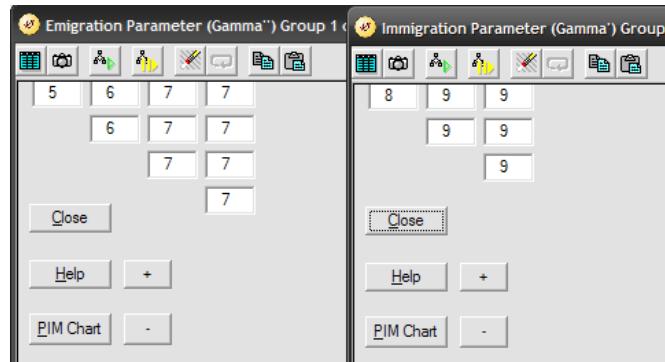
If you've read the preceding text carefully, you'll recognize that in fact (i) this represents a Markovian model for the γ parameters, and (ii) without constraints, there will be identifiability problems for S and γ for this model.

We can confirm this by running the model, and looking at the estimates for S and γ from this model:

Parameter	Estimate	Standard Error	95% Confidence Interval	
			Lower	Upper
1:S	0.6907290	0.0339920	0.6204952	0.7531361
2:S	0.7912726	0.5114074	0.0086940	0.9993901
3:S	0.9669460	8.2556539	0.3947704E-218	1.0000000
4:S	0.9693614	12.133730	0.1759953E-306	1.0000000
5:Gamma''	0.1943705	0.0398493	0.1277902	0.2843316
6:Gamma''	0.3195660	0.4399406	0.0088249	0.9612008
7:Gamma''	0.3427122	5.6118584	0.3243531E-021	1.0000000
8:Gamma''	0.3275440	8.4173046	0.1438525E-032	1.0000000
9:Gamma'	0.2007573	0.3265811	0.0046286	0.9313578
10:Gamma'	0.4238489	6.0838579	0.4571182E-021	1.0000000
11:Gamma'	0.5208728	13.362561	0.2877972E-045	1.0000000

We see that the estimates for the last two S and γ parameters are completely confounded. Now, let's see what happens to the estimates if we apply the constraints $\gamma'_k = \gamma'_{k-1}$, and $\gamma''_k = \gamma''_{k-1}$? As mentioned earlier, these constraints are necessary to make S and the remaining γ parameters identifiable. How do we set these constraints?

Here, we'll use a simple PIM-based approach. Here are the modified PIMs for γ'' and γ' , respectively:



Make sure you understand what we've done in the PIMs. We've set the last two parameters equal to each other for both γ'' (parameter index 7) and γ' (parameter index 9). This constraint should allow us to estimate γ''_2 (parameter index 5) and γ''_3 (parameter index 6), and γ'_3 (parameter index 7).

If we fit this model to the data, we see that the estimates of S and γ (shown at the top of the next page) are all reasonable, and quite close to the true underlying parameter values used in the simulation ($S_1 = 0.7, S_2 = 0.8, S_3 = 0.9, S_4 = 0.8; \gamma''_2 = 0.2, \gamma''_3 = 0.3; \gamma''_4 = 0.3, \gamma''_5 = 0.2$ and $\gamma'_3 = 0.2$). Remember that we've achieved this 'identifiability' by applying constraints on the terminal pairs of γ parameters – not only may there be no good biological justification for imposing this constraint, but the estimates of the constrained γ (parameter index 7, representing the constraint $\gamma''_4 = \gamma''_5$, and parameter index 9, representing the constraint $\gamma'_4 = \gamma'_5$) are not biologically interpretable.

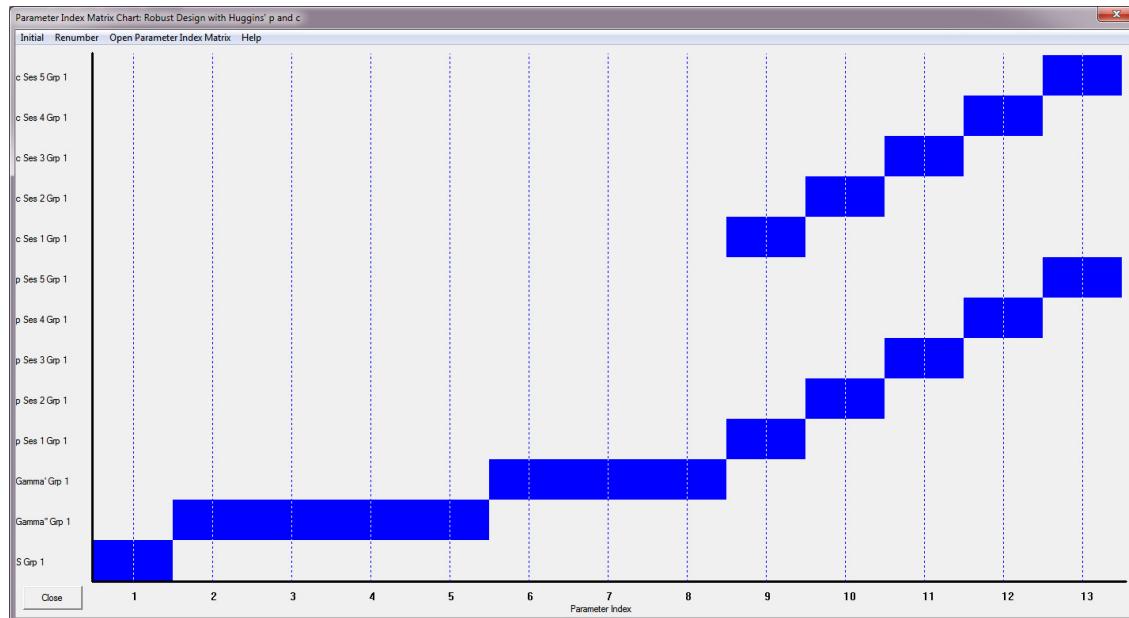
What if instead of constraining γ we'd applied a constraint to the survival parameter S ? For example, what if we constrained S to be constant over time? As you'll recall from earlier chapters, constraining one parameter can often eliminate confounding with other parameters, and in the process, make them identifiable. For example, in a simple $\{\varphi_t, p_t\}$ live mark-recapture model, the terminal φ and p parameters

Parameter	Estimate	Standard Error	95% Confidence Interval	
			Lower	Upper
1:S	0.6910108	0.0107492	0.6695554	0.7116730
2:S	0.7868335	0.0155367	0.7548014	0.8157027
3:S	0.9001962	0.0247884	0.8400535	0.9393568
4:S	0.9232708	0.0304892	0.8380905	0.9654835
5:Gamma''	0.1946990	0.0131690	0.1701757	0.2218117
6:Gamma''	0.3157268	0.0181493	0.2812771	0.3523274
7:Gamma''	0.2939742	0.0220538	0.2526702	0.3389677
8:Gamma'	0.1979316	0.0376100	0.1342830	0.2819243
9:Gamma'	0.3700688	0.0533513	0.2727952	0.4791730

are confounded, whereas if you fit model $\{\varphi, p_t\}$ (i.e., constrain φ to be constant over time), all of the encounter probabilities p_t are estimable, including the terminal parameter. Of course, you would still want to have a good prior motivation to apply the constraint.

So, for our present analysis, what happens to our estimates if we (i) constrain S to be constant over time, and (ii) ‘remove’ the $\gamma'_k = \gamma'_{k-1}$ and $\gamma''_k = \gamma''_{k-1}$ constraints?

Here is the PIM chart corresponding to this model – note that the indexing for γ'' is now $2 \rightarrow 5$ (whereas for the constrained model, it was $2 \rightarrow 4$), and for γ' , the indexing is from $6 \rightarrow 8$ (instead of $6 \rightarrow 7$ for the constrained model).



The estimates from fitting this model with constant survival to the data are shown at the top of the next page. We see that in fact all of the γ'' parameters are now estimable, as are the first two estimates for γ' . The estimates qualitatively match the true underlying parameter values – differences reflect the fact that in the generating model used to simulate the data, survival S was time-dependent – here we are constraining it to be constant over time, which affects our estimates of other parameters.

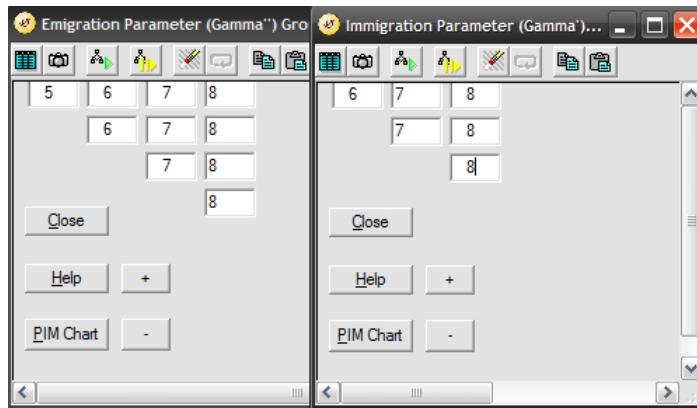
Real Function Parameters of {Markovian - constant S constraint}				
Parameter	Estimate	Standard Error	95% Confidence Interval	
			Lower	Upper
1:S	0.7597886	0.0054666	0.7489112	0.7703392
2:Gamma''	0.2304095	0.0129935	0.2059320	0.2568553
3:Gamma''	0.3174525	0.0161458	0.2866817	0.3499059
4:Gamma''	0.2225764	0.0202414	0.1854286	0.2647474
5:Gamma''	0.1301263	0.0231132	0.0911181	0.1824813
6:Gamma'	0.2657133	0.0376899	0.1985947	0.3457299
7:Gamma'	0.3057709	0.0332918	0.2446487	0.3745908
8:Gamma'	0.1893485E-012	0.7905384E-007	-0.1549453E-006	0.1549457E-006

Let's continue fitting the models in our candidate model set, assuming that S is time-dependent (go ahead and delete the model we just ran with S held constant from the browser – we ran that model just to demonstrate that you could achieve identifiability by holding S constant). We'll fit the '*Random movement*' model next. Recall that for a '*Random movement*' model, which is essentially the 'classical' robust design, we apply the constraint $\gamma_i'' = \gamma_i'$.

Also remember that without additional constraints, the parameters γ_k'' , γ_k' , and S_{k-1} are all confounded. While you could set some constraints to 'pull them apart', in practice it is often easier to forgo the constraint – in that case, you would simply ignore the estimates of S_{k-1} , γ_k' , and γ_k'' . Estimates of the remaining parameters would be unbiased.

Specifying the '*Random movement*' model is straightforward, but remember that there is one more γ'' than γ' parameter. In this case, there is no γ_2' parameter corresponding with γ_2'' , so we apply the constraint to the γ parameters for primary occasions 3 and 4 only.

Again, this is most easily accomplished by modifying the PIMs for γ'' and γ' , respectively:



Run this model and add the results to the browser.

If you look at the real estimates from the '*Random movement*' model (shown at the top of the next page) you see that only the final S and γ parameters are confounded.*

* You might have noticed that the structure of the '*Random movement*' model, where we set $\gamma' = \gamma''$, is strictly analogous to model M_t (i.e., model $\{p_t = c_t\}$) for closed abundance models (Chapter 15).

Real Function Parameters of {Random movement}				
Parameter	Estimate	Standard Error	95% Confidence Interval	
			Lower	Upper
1:S	0.6954807	0.0107545	0.6740017	0.7161404
2:S	0.7742020	0.0135466	0.7465524	0.7996448
3:S	0.8784287	0.0184392	0.8374276	0.9101982
4:S	0.7649371	40.554635	0.3355783E-191	1.0000000
5:Gamma''	0.1998747	0.0131876	0.1752739	0.2269781
6:Gamma''	0.2877120	0.0151780	0.2589015	0.3183513
7:Gamma''	0.2918787	0.0202238	0.2538733	0.3330340
8:Gamma''	0.1478351	45.179191	0.9413177E-306	1.0000000

Finally, the ‘No movement’ model. Recall that to fit this model, we fix $\gamma'_i = 1$ and $\gamma''_i = 0$ over all occasions. We can do this easily by using the ‘Fix parameters’ button in the ‘Run numerical estimation’ window. Since we just finished building the ‘Random movement’ model, we can run the ‘No movement’ model simply by fixing all of the γ parameters in the ‘Random movement’ model (parameters 5 → 8) to either 1 or 0 (for γ' and γ'' respectively). Go ahead and fix the γ ’s, run the ‘No movement’ model, and add the results to the browser:

Model	AICc	Delta AICc	AICc Weight	Model Likelihood	No. Par.	Deviance	-2Log(L)
{Markovian - standard constraints}	39171.3723	0.0000	0.96167	1.0000	14	94397.7900	39143.3400
{Random movement}	39177.8170	6.4447	0.03833	0.0399	12	94408.2430	39153.7930
{No movement model}	40028.9248	857.5525	0.00000	0.0000	9	95265.3610	40010.9110

In looking at our results, we conclude that the ‘Markovian movement’ model has by far the most support in the data among our 3 candidate models. This should not be surprising – a Markovian model was the generating model for the simulated data.

Using the design matrix in the RD – simple example revisited...

In the preceding, we built the models using PIMs. How would we build these models using the design matrix (DM)? We start by considering the PIM structure for a model with full time-dependence in S and γ (i.e., a Markovian emigration model), with annual variation in p (in fact, this is the generating model used to simulate the data we considered in the preceding example). The PIM chart corresponding to this structure is the one shown at the bottom of p. 16. Again, this is the model without the constraints needed to make the γ parameters estimable in the Markovian model.

While this model without constraints on the γ parameters is not a good model for inference (since, many of the S and γ parameters will be confounded), it is, structurally, the starting point for all of the models we’re interested in building (at the least, the ‘Markovian’, ‘random’, and ‘no movement’ models). This is analogous to using a model with full time-dependence and interaction for the p and c parameters in a closed population abundance model (Chapter 15) as the starting point for building other constrained, reduced parameters model.

By now, you should realize that there are a number of ways to build the DM corresponding to parameter structure for this model. One way, which might occur to you to be the ‘default’ approach based on ‘intercept offset coding’, is shown at the top of the next page.

Here, we assume that we’re going to model each of the structural parameters in the model (S, γ', γ'', p) independently of each other. Meaning, each parameter will have its own intercept (as shown). While there is nothing wrong with this, it makes it somewhat more difficult to build models where we want

Design Matrix Specification (B = Beta)																
B1 int-S	B2 S+1	B3 S+2	B4 S+3	B5 g-dp-int	B6 g-dp+1	B7 g-dp+2	B8 g-dp+3	B9 g-p-int	B10 g-p+2	B11 g-p+3	Parm	B12 p-int	B13 p+1	B14 p+2	B15 p+3	B16 p+4
1	1	0	0	0	0	0	0	0	0	0	1:S	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	2:S	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	3:S	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	4:S	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0	0	5:Gamma"	0	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	6:Gamma"	0	0	0	0	0
0	0	0	0	1	0	0	1	0	0	0	7:Gamma"	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	8:Gamma"	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	9:Gamma'	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	10:Gamma'	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	11:Gamma'	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	12:p Session 1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	13:p Session 2	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	14:p Session 3	1	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	15:p Session 4	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	16:p Session 5	1	0	0	0	0

(or need) to specify particular relationships between 2 or more of the parameters. For example, there is no simple modification of this DM which will let you build a ‘Random movement’ model, where $\gamma' = \gamma''$.

Is there a more flexible approach? In fact, you might recall from our development of the DM for closed population abundance models (Chapter 15 – section 15.6) that a straightforward approach is to consider each of the parameters you want to model ‘together’ (i.e., as being related to each other in some fashion) as different levels of a putative ‘group’ factor, using a common intercept for these parameters. [In fact, you may recall that we first introduced this concept back in Chapter 7 with respect to ‘age’ and ‘time since marking’ models, and saw it again in Chapter 15 when using a common intercept for p and c parameters in closed population abundance models]. We start by specifying a putative parameter ‘group’ for the γ parameters – we’ll call it ‘GG’ (for ‘ γ -group’).

Next, to help us keep track of what we’re doing, we write out the linear model corresponding to the γ parameters shown in the PIM chart on p. 18. We know that not all parameters are identifiable without constraints, but it represents our most general parametrization for γ , which we will constrain to build our 3 candidate models.

Here is the linear model corresponding to the γ parameters shown in the PIM chart:

$$\begin{aligned}\gamma' &= \text{INTCPT} + \text{GG} + \text{TIME} + \text{GG} \cdot \text{TIME} \\ &= \beta_1 + \beta_2(\text{GG}) + \beta_3(\text{T}_1) + \beta_4(\text{T}_2) + \beta_5(\text{T}_3) + \beta_6(\text{GG} \cdot \text{T}_2) + \beta_7(\text{GG} \cdot \text{T}_3)\end{aligned}$$

We see there are 7 terms in the linear model, which correctly matches the 7 parameters for γ specified in the PIM chart. Note that we model only ‘plausible interactions’. Since there is no γ'_1 parameter, then no interaction is specified for interval coded by T_1 . So, the interactions between ‘GG’ and time begin with the second interval (β_6 in the linear model).

Shown at the top of the next page is what this linear model – for the γ parameters – would look like coded in the DM (note that we’ll leave the coding for the other parameters S and p the same). Notice that it is identical in structure to many of the age and closed population abundance models we saw in earlier chapters, for the same reason: two parameters offset from a common intercept, where one of the parameters does not occur in the first time interval.

Parm	B5: g-intcpt	B6: gg	B7: t1	B8: t2	B9: t3	B10: gg*t2	B11: gg*t3
1:S	0	0	0	0	0	0	0
2:S	0	0	0	0	0	0	0
3:S	0	0	0	0	0	0	0
4:S	0	0	0	0	0	0	0
5:Gamma"	1	1	1	0	0	0	0
6:Gamma"	1	1	0	1	0	1	0
7:Gamma"	1	1	0	0	1	0	1
8:Gamma"	1	1	0	0	0	0	0
9:Gamma'	1	0	0	1	0	0	0
10:Gamma'	1	0	0	0	1	0	0
11:Gamma'	1	0	0	0	0	0	0

Given this DM, how would we modify it to construct the 3 models we constructed in the preceding section using the PIM approach? We'll start with the '*Markovian movement*' model. Recall for a model where movement is modeled as Markovian, we generally need to set (i) $\gamma''_k = \gamma''_{k-1}$ and (ii) $\gamma'_k = \gamma'_{k-1}$. How would we set these constraints in the DM? The key is in realizing that by setting $\gamma''_k = \gamma''_{k-1}$ and $\gamma'_k = \gamma'_{k-1}$ we are, in effect, 'merging' the final two time intervals. In other words, instead of having 4 time intervals, we now have 3.

Now, have another look at the part of the DM relating to the γ parameters (shown above). Note that column B9 (labeled t3) specifies the 3rd time interval. So, to make time interval 3 equivalent to time interval 4 (which we coded by convention as the reference interval), we simply delete the DM column corresponding to the third interval (column B9 in the preceding). But, if we delete the that column, then we also need to delete any interaction column involving that time interval (column B11).

After doing that, here is what the modified DM for the '*Markovian movement*' model for γ looks like:

B5: g-intcpt	B6: gg	B7: t1	B8: t2	B9: gg*t2	Parm
0	0	0	0	0	1:S
0	0	0	0	0	2:S
0	0	0	0	0	3:S
0	0	0	0	0	4:S
1	1	1	0	0	5:Gamma"
1	1	0	1	1	6:Gamma"
1	1	0	0	0	7:Gamma"
1	0	0	0	0	8:Gamma"
1	0	0	1	0	9:Gamma'
1	0	0	0	0	10:Gamma'
1	0	0	0	0	11:Gamma'

If you run this model, add the results to the browser, and look at the real parameter estimates, you'll see that they match exactly the results from the '*Markovian movement*' model built earlier using the PIM approach.

Next, we'll consider the '*Random movement*' model. Recall for this model, we set $\gamma' = \gamma''$. This is analogous to setting $p = c$ in a closed population abundance model. By setting $\gamma' = \gamma''$, we are in effect eliminating the difference between the groups. So, first retrieve the 'starter' model that we used to construct the Markovian model (the DM for the γ parameters for this model is shown at the top of this page). All we need to do next is (i) delete the 'GG' column (B6), and (ii) delete any interaction column involving 'GG' (in this case, columns B10 and B11, labeled 'GG.t2' and 'GG.t3').

Here is the modified DM for γ for the ‘Random movement’ model:

B5: g-int	B6: t1	B7: t2	B8: t3	Parm
0	0	0	0	1:S
0	0	0	0	2:S
0	0	0	0	3:S
0	0	0	0	4:S
1	1	0	0	5:Gamma''
1	0	1	0	6:Gamma''
1	0	0	1	7:Gamma''
1	0	0	0	8:Gamma''
1	0	1	0	9:Gamma'
1	0	0	1	10:Gamma'
1	0	0	0	11:Gamma'

Again, if you run this model, and add the results to the browser – you’ll see that they match the results from the ‘Random movement’ model built using the PIM approach. Note that by simply modifying the DM we built for the ‘Markovian movement’ model, we are retaining the $\gamma''_k = \gamma''_{k-1}$ and $\gamma'_k = \gamma'_{k-1}$ constraints. Doing so (or not) does not affect the overall model fit, but does influence the identifiability of some parameters (in particular, the final estimate for survival S).

Finally, for the (null) ‘no movement’ model, we want to set $\gamma' = 1$, and $\gamma'' = 0$. Starting from the DM we just specified for the ‘Random movement’ model, all we need to do is (i) delete the time columns (B7 → B9), and (ii) fix parameters 5 → 8 to 0, and parameters 9 → 11 to 1 when we run the model. Results from this model should match those from the ‘no movement’ model built using PIMs.

16.6.2. Closed robust design – more complex worked example

In the preceding, we used PIMs, and a PIM chart, to construct the various models in our candidate model set. We also demonstrated (and reinforced) the notion of parameter constraints that are often necessary to make various parameters estimable. Recall in particular that for a model with Markovian movement, setting the constraints $\gamma''_{k-1} = \gamma''_k$ and $\gamma'_{k-1} = \gamma'_k$ (where k is the number of primary sampling occasions) makes the resulting 3 parameters estimable. However, as you might imagine, there is another approach which is often preferred, since it makes various parameters estimable, while avoiding constraints which work, but may have little biological meaning, or justification. This approach involves constraining various estimates to be functions of one or more covariates.

Consider the following scenario – suppose that only individuals in the breeding condition are available for encounter (this is quite plausible – for many taxa, only reproductively active individuals are ever encountered). Non-breeding individuals often do not return to the breeding site, and are thus not available for encounter). Suppose that in general, for some species, if the climatic conditions are favorable, the tendency of all individuals to breed is increased, relative to a year with harsher climatic conditions, where more individuals opt out of breeding. Thus, we would anticipate that in general in a ‘good’ year, $(1 - \gamma')$ and $(1 - \gamma'')$ will generally be greater than γ' and γ'' . The reverse would generally be true in a ‘poor’ year.

For this example, we simulated 15 occasions worth of data – 5 primary sampling occasions, each with 3 secondary samples. Primary samples 1, 2 and 4 were classified as ‘good’ years, whereas primary samples 3 and 5 were taken in ‘poor’ years. In ‘good’ years, $\gamma'_g = 0.5$ and $\gamma''_g = 0.1$. In ‘poor’ years,

$\gamma'_p = 0.7$ and $\gamma''_p = 0.25$. These values were chosen to reflect the basic expectation that in ‘good’ years, individuals that were breeders the previous year tend to remain in breeding state, whereas individuals that were not breeding the year before tend to become breeders. The reverse is likely true (in many cases) in ‘poor’ years. We also assumed that survival is marginally lower in ‘poor’ years than in ‘good’ years ($S_g = 0.8 > S_p = 0.7$). Finally, we also assumed that encounter effort tends to be lower in ‘poor’ years (perhaps for some logistical reasons related to the poorer weather), but that $p_i^* = c_i$ in all years. We set $p_g^* = c_g = 0.5$, and $p_p^* = c_p = 0.3$. We simulated a data set with 2,000 individuals captured, marked and released alive on the first occasion. We assumed closure within each primary sampling period, and no net immigration of new individuals into the population on any subsequent occasion (thus, expected population size N should decline over time). The simulated data are contained in the file **rd_complex1.inp**.

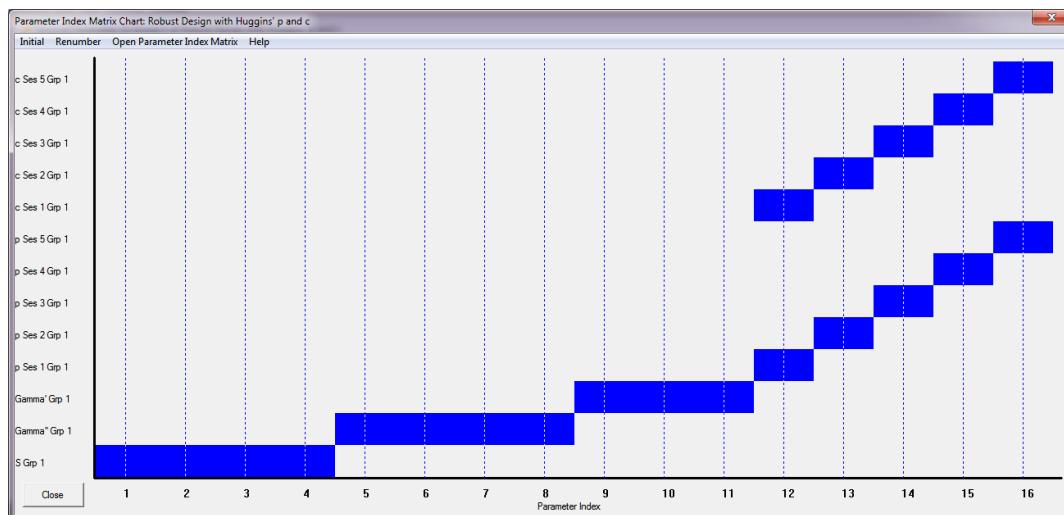
Now, let’s proceed to analyze these data – with the intent of demonstrating that use of covariates can make it possible to estimate various parameters without relying on equality constraints as described earlier. We’ll assume that our model set consists of 2 models:

1. Markovian, no covariates – simple time variation in S , γ , and $p = c$
2. Markovian, with covariates used to explain temporal variation in S , γ , and $p = c$.

Now, in this example, the covariate is a dichotomous indicator variable (‘good’ year or ‘poor’ year). As such, this example problem is analogous to the (by now) familiar European dipper ‘flood’ analysis. You may recall from Chapter 6 that there are two ways to approach this type of analysis. We could, in fact, use a PIM-only approach for some models, by coding ‘good’ and ‘poor’ directly into the PIMs for various parameters (see section 6.7 in Chapter 6). However, we also recall that ultimately this limits the types of models we want to build – more generally, we’d like to use a design matrix approach, since it will (ultimately) give us complete flexibility over the types of models we build. So, that is the approach we’ll employ here for our model with covariates for ‘good’ and ‘poor’ years.

Start **MARK**, and access the **rd_complex1.inp** file. Select ‘robust design | Huggins’ **p** and **c**’ as the model type – 15 total occasions, with 5 primary occasions each consisting of 3 secondary samples. To start, we’ll build the unconstrained model – time variation in S , γ , and $p = c$. We can do this most easily by making use of the PIM chart.

Go ahead and bring up the PIM chart, and modify it so it looks like the following:



This PIM chart is similar to the PIM chart used in the preceding (simpler) example – the only major difference is that now the ‘blue boxes’ for S , γ'' and γ' are all time-dependent. Recall that this has implications for estimability of several parameters.

Now, we recall from earlier sections of this chapter that for time-dependent robust design models, not all parameters are estimable without some constraints. Specifically, we would need to set the constraints $\gamma''_4 = \gamma''_5$ and $\gamma'_4 = \gamma'_5$. However, for purposes of demonstrating the necessity of these constraints, let’s first run the model without imposing the constraints. We see from the following listing of parameter estimates that indeed, none of the survival or γ parameters are estimable (they all have completely unrealistic SE or 95% CI).

Real Function Parameters of {no constraints - no covars - PIM model - sin link}				
Parameter	Estimate	Standard Error	95% Confidence Interval	
			Lower	Upper
1:S	0.8089838	0.0000000	0.8089838	0.8089838
2:S	0.7871476	0.0000000	0.7871476	0.7871476
3:S	0.5925969	0.0000000	0.5925969	0.5925969
4:S	0.8186455	0.0000000	0.8186455	0.8186455
1:....1....2....3....4....5....6....7:J....8....9:..				
5:Gamma''	0.1076813	0.0000000	0.1076813	0.1076813
6:Gamma''	0.0962944	0.0000000	0.0962944	0.0962944
7:Gamma''	0.1426107	0.0000000	0.1426107	0.1426107
8:Gamma''	0.0085969	0.0000000	0.0085969	0.0085969
9:Gamma'	0.7026864	0.0000000	0.7026864	0.7026864
10:Gamma'	0.4259819	0.0000000	0.4259819	0.4259819
11:Gamma'	0.0601455	0.0000000	0.0601455	0.0601455
====>				

Now, let’s re-run this same model, after applying the constraints $\gamma''_4 = \gamma''_5$ and $\gamma'_4 = \gamma'_5$. In the preceding example, we applied these constraints by directly modifying the PIMs for both γ parameters. However, this is not necessary – we can apply these constraints using the design matrix, which we want to build anyway, for the purposes of constraining our estimates to be functions of ‘good’ or ‘poor’ years.

First, let’s build the design matrix (DM) which corresponds to the PIM chart shown on the preceding page – since our models set consists of only 2 candidate models (Markovian with and without covariates), we’ll use the ‘default’ coding which treats γ' and γ'' separately (i.e., each parameter has its own intercept).

B1 S-int	B2 t1	B3 t2	B4 t3	B5 gdp-int	B6 t1	B7 t2	B8 t3	B9 gp-int	B10 t2	B11 t3	Parm	B12 p-int	B13 t1	B14 t2	B15 t3	B16 t4
1	1	0	0	0	0	0	0	0	0	0	1:S	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	2:S	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	3:S	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	4:S	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0	0	5:Gamma''	0	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	6:Gamma''	0	0	0	0	0
0	0	0	0	1	0	0	1	0	0	0	7:Gamma''	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	8:Gamma''	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	9:Gamma'	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	1	10:Gamma'	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	11:Gamma'	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	12:p Session 1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	1	13:p Session 2	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	14:p Session 3	1	0	0	1	0
0	0	0	0	0	0	0	0	0	0	1	15:p Session 4	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	16:p Session 5	1	0	0	0	0

Now, we want to modify this starting DM to constrain $\gamma_4'' = \gamma_5''$ and $\gamma_4' = \gamma_5'$. Recall from the first, simpler example we worked through in this chapter that at present, the γ parameters for occasion k and $k - 1$ are coded as separate time intervals. Thus, to apply the necessary constraint, we simply modify the DM so that the final two time intervals are treated as a single time step.

	1	2	3	4	5	6
5:Gamma"	1	1	0	0	0	0
6:Gamma"	1	0	1	0	0	0
7:Gamma"	1	0	0	0	0	0
8:Gamma"	1	0	0	0	0	0
9:Gamma'	0	0	0	1	1	0
10:Gamma'	0	0	0	1	0	0
11:Gamma'	0	0	0	1	0	0
12:n Session 1	n	n	n	n	n	1

If you run this model, you'll see that now, all of the non-constrained γ parameters, and S are estimable (i.e., have reasonable standard errors, and are qualitatively close to the true parameter values). We can confirm this by modifying the PIMs directly (as in the preceding example). You should get precisely the same estimates as you just did using the design matrix. You will also note that the deviance of this 'constrained' model is identical to that for the unconstrained model we fit on the previous page (since the point estimates for the parameters are identical – all that has changed are the SE's).

But, again, we achieved 'estimability' at the cost of imposing some necessary, but perhaps not particularly 'biologically meaningful' constraints. Remember that for this example, primary samples 1, 2 and 4 were classified as 'good' years, whereas primary samples 3 and 5 were taken in 'poor' years. In 'good' years, $\gamma_g' = 0.5$ and $\gamma_g'' = 0.1$. In 'poor' years, $\gamma_p' = 0.7$ and $\gamma_p'' = 0.25$. Given this structure, it makes little *biological* sense to constrain $\gamma_4'' = \gamma_5''$ and $\gamma_4' = \gamma_5'$. Although these constraints did make the non-constrained γ parameters estimable, there would be reason to be concerned about possible bias in the unconstrained estimates (relative to the true values).

It would seem to be more appropriate to modify the DM to account for variation between 'good' and 'poor' years. Let '1' be the dummy variable we use to code for a 'good' year, and '0' be the dummy variable coding for a 'poor' year. Recall that primary sessions 1, 2 and 4 occurred in 'good' years, while primary sessions 3 and 5 occurred in 'poor' years. Recall that we assume that S , γ , and $p = c$ are all functions of whether or not a year was classified as 'good' or 'poor'.

Start by retrieving the model constructed using a DM *without* the logical constraints $\gamma_4'' = \gamma_5''$ and $\gamma_4' = \gamma_5'$. We'll begin by modifying the coding for the survival parameter S first. In order to do this, we need to decide on whether or not S or γ over the interval from (i) to $(i + 1)$ are functions of whether or not the environment is 'good' or 'poor' at time (i) .

For this example, we'll assume that since



that the estimate for parameter θ_i is a function of the state of the environment at the start of the interval from (i) to $(i + 1)$. So, for example, S_1 , S_2 , and S_4 reflect 'good' years, whereas S_3 reflects a 'poor' year. In contrast, for parameters estimated *at* a particular sampling occasion (e.g., c , p , N), the estimate for θ_i reflects the conditions at sampling occasion (i) .

We'll start by modifying the part of the DM corresponding to survival.

Design Matrix Specification (B = Beta)										
B0	B1	B2	B3	B4	B5	B6	B7	B8	Parm	E
1	1	0	0	0	0	0	0	0	1:S	0
1	1	0	0	0	0	0	0	0	2:S	0
1	0	0	0	0	0	0	0	0	3:S	0
1	1	0	0	0	0	0	0	0	4:S	0

The first column (labeled B0) is the intercept, while the second column (labeled B1) is the coding for 'good' or 'poor' years. Again, note that in our coding we are assuming that the conditions ('good' or 'poor') at primary occasion (i) determine the probability of surviving from occasion (i) → ($i + 1$).

Now, what about γ' and γ'' ? Here, all we need to remember is that there is one fewer occasion for the γ' parameter (for reasons discussed earlier in this chapter). The primary challenge, then, is to keep track of which row refers to which occasion, for each of the two γ parameters. For $\gamma_2'' \rightarrow \gamma_5''$, we have 'good', 'good', 'poor' and 'good', whereas for $\gamma_3' \rightarrow \gamma_5'$ we have 'good', 'poor' and 'good'.

Here is the completed DM for the γ parameters:

1	1	0	0	0	0	5:Gamma"	0
1	1	0	0	0	0	6:Gamma"	0
1	0	0	0	0	0	7:Gamma"	0
1	1	0	0	0	0	8:Gamma"	0
0	0	1	1	0	0	9:Gamma'	0
0	0	1	0	0	0	10:Gamma'	0
0	0	1	1	0	0	11:Gamma'	0

Finally, we modify the DM for the $p = c$ parameters. Remember that we're assuming that $p_i = c_i$ is equal to the conditions at sampling occasion (i). Thus,

12:p Session 1	1	1
13:p Session 2	1	1
14:p Session 3	1	0
15:p Session 4	1	1
16:p Session 5	1	0

Not surprisingly, when we run this model and add the results to the browser (shown below), we see it has most of the support in the data (this is a good thing, since this is the 'true' model under which the data were simulated in the first place).

Model	AICc	Delta AICc	AICc Weight	Model Likelihood	No. Par.	Deviance	-2Log(L)
{no constraints - with covars - DM model}	25904.6201	0.0000	0.51213	1.0000	8	58656.6210	25888.6010
{with constraints - no covars - DM model}	25906.2843	1.6642	0.22284	0.4351	13	58648.2560	25880.2360
{no constraints - no covars - DM model}	25908.1348	3.5147	0.08834	0.1725	14	58648.0990	25880.0790
{no constraints - nocovars - PIM model - sin link}	25908.1348	3.5147	0.08834	0.1725	14	58648.0990	25880.0790
{no constraints - nocovars - PIM model - logit link}	25908.1348	3.5147	0.08834	0.1725	14	58648.0990	25880.0790

However, what is of greater interest here is the influence of adding covariates to the DM on the

estimability of the various parameters in the model. As shown below, all of the parameters are estimable (dichotomized between 'good' and 'poor' years). It is worth noting that we assume the survival process is the same for those that are or are not available at any given time. We cannot derive a separate estimate of survival for individuals in and outside of the sample – to do so requires different approaches, discussed elsewhere.

Parameter	Estimate	Standard Error	95% Confidence Interval	
			Lower	Upper
1:S	0.8340096	0.0323643	0.7606236	0.8882042
2:S	0.8340096	0.0323643	0.7606236	0.8882042
3:S	0.7557498	0.1242223	0.4527721	0.9204525
4:S	0.8340096	0.0323643	0.7606236	0.8882042
5:Gamma''	0.1376021	0.0324863	0.0853393	0.2143697
6:Gamma''	0.1376021	0.0324863	0.0853393	0.2143697
7:Gamma''	0.3195608	0.1144854	0.1433486	0.5686070
8:Gamma'	0.1376021	0.0324863	0.0853393	0.2143697
9:Gamma'	0.7412164	0.1401822	0.4061005	0.9230630
10:Gamma'	0.7164796	0.1480153	0.3772853	0.9133481
11:Gamma'	0.7412164	0.1401822	0.4061005	0.9230630
12:p Session 1	0.4967553	0.0059596	0.4850784	0.5084357
13:p Session 2	0.4967553	0.0059596	0.4850784	0.5084357
14:p Session 3	0.3024502	0.0086419	0.2857853	0.3196519
15:p Session 4	0.4967553	0.0059596	0.4850784	0.5084357
16:p Session 5	0.3024502	0.0086419	0.2857853	0.3196519

16.7. The multi-state closed RD

In section 16.3.1 we briefly described the analogy between a model including temporary emigration from a single study area and a multi-state model with two states. I will illustrate this in more detail in this section, for two reasons. First, the multi-state closed robust design (hereafter, MSCRD) model, initially presented by Nichols & Coffman (1999), provides much more flexibility than either the original robust design model (which only permits two states, one of which must be unobservable) or the multi-state model (which lacks the extra information available from multiple secondary capture periods). However, as mentioned earlier, with flexibility comes complexity, and you will see that setting up the multi-state robust design model in **MARK** can be very involved (and tedious). Second, this section provides a segue to the multi-state open robust design (MSORD). In **MARK**, for the open robust design there is no simpler alternative to the full multi-state version. Again, the flexibility of this model will compensate for the complexity.

Between Chapter 10 and this chapter up to this point, the pieces of the MSCRD have already been explained individually. For someone familiar with the MS model of Chapter 10, the simplest way to view the MSCRD model is to note that each time capture probability p_i^s for state s appears in a MS model, it is replaced with p_i^{*s} . As in previous sections, if there are three secondary capture periods for primary period (i), then the effective capture probability for state s in primary capture period (i) might be $p_i^{*s} = 1 - (1 - p_{i1}^s)(1 - p_{i2}^s)(1 - p_{i3}^s)$. The easiest way to illustrate the relationship between the classical robust design model and the MSCRD model is through an example. For simplicity we'll use the simple robust design example from section 16.6.1.

16.7.1. multi-state closed RD – simple worked example

The simple example discussed in section 16.6.1 involved a hypothetical study consisting of 4 primary periods of interest. Capture effort for each of these primary periods consisted of 3 secondary capture periods, conducted over a sufficiently short period of time that it is reasonable to assume the group of

animals in the study area did not change (no deaths or births or movement in or out). This constitutes a robust design. The data were generated from a population with time-varying survival ($S_1 = 0.7, S_2 = 0.8, S_3 = 0.9, S_4 = 0.8$). Capture probability varied across primary periods ($p_{1j} = 0.5, p_{2j} = 0.6, p_{3j} = 0.6, p_{4j} = 0.5, p_{5j} = 0.7$), but not within primary period, and recapture probability within a primary period was the same as initial capture probability (i.e., no trap effect; model M_0 – see Chapter 15). All of these parameters are also found in the MSCRD model. Notation changes when we consider the transition (in this case movement) parameters (i.e., γ and ψ).

Relating the two sets of notation to the parameter values chosen, recall (from earlier in this chapter) that

$$\begin{aligned}\gamma' &\equiv \psi^{UU} \\ 1 - \gamma' &\equiv \psi^{UO} \\ \gamma'' &\equiv \psi^{OU} \\ 1 - \gamma'' &\equiv \psi^{OO}\end{aligned}$$

where superscript ‘O’ means ‘observable’, or *within the study area*, and superscript ‘U’ means ‘unobservable’, or *outside the study area*.

In this example $\psi_1^{OU} \equiv \gamma'' = 0.2, \psi_2^{OU} \equiv \gamma_3'' = \psi_3^{OU} \equiv \gamma_4'' = 0.3, \psi_4^{OU} \equiv \gamma_5'' = 0.2$, and $\psi_2^{UU} \equiv \gamma_3' = 0.2, \psi_3^{UU} \equiv \gamma_4' = 0.4, \psi_4^{UU} \equiv \gamma_5' = 0.3$. The analogy between the classic closed RD and the MS equivalent will become even clearer after running this in **MARK** using the MSCRD model and comparing it to the output from the original ‘classic’ analysis.

For this exercise, start by making a copy of ‘**rd_simple1.inp**’ and call it ‘**MS_rdsimple1.inp**’. To use the MSCRD model simply open the **MARK** screen for developing new models and click on ‘**Closed Robust Design Multi-state**’ (almost the last model listed). For consistency with our previous run with these data, choose the ‘**Huggins p and c**’ option.

As in the previous example, after selecting the input file, specify 15 encounter occasions (remember there were 5 primary periods, each consisting of 3 secondary capture occasions). Click on ‘**Easy Robust Times**’ and specify 5 primary periods. It will give you a new screen that happens to have the correct allocation of capture occasions across primary periods (3 in each). If in doubt, check back to section 16.6.1 where some of these steps were first introduced.

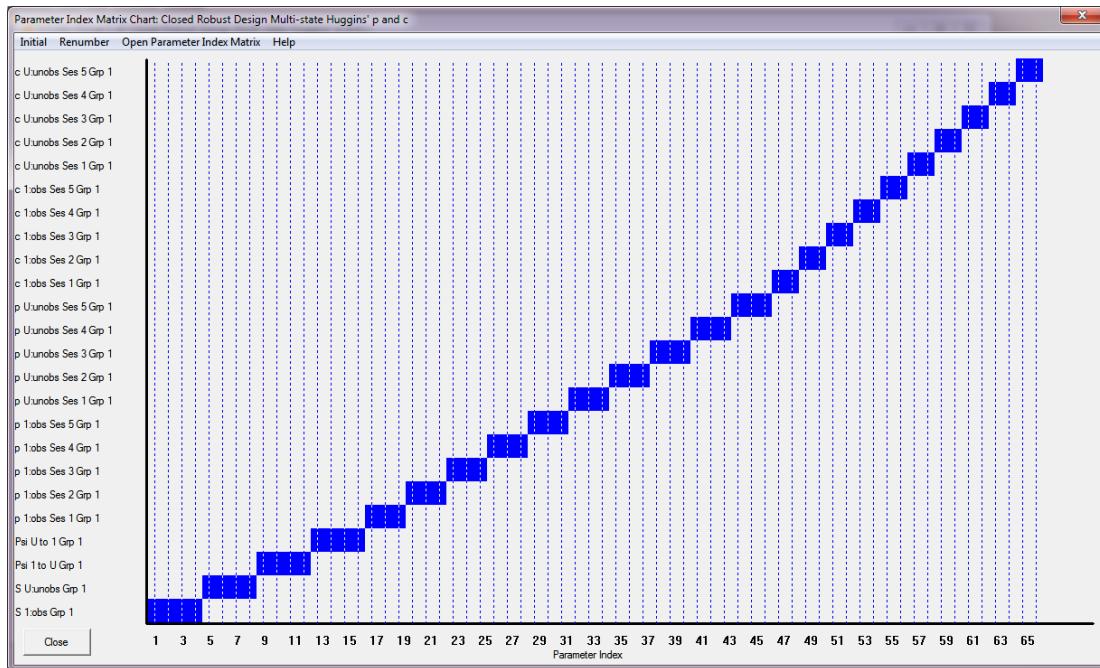
Up to this point the process has been identical to using the ‘**Robust Design**’ option in **MARK**. Now we run into the first difference, and complication. Because this is a *multi-state* model, you see that the ‘**State**’ and ‘**Enter State Names**’ sections are now active. From Chapter 10 recall that you must specify for each state the code that will be used in the capture history to denote capture in that state. You also have the option of specifying a label for that state that might be more descriptive than the code alone. The default value of 2 states is appropriate for this example, because we have two states in this example (‘0’ = observable = available for capture in the study area, ‘U’ = unobservable = outside the study area).

Click on ‘**Enter State Names**’ and you will see the default codes of ‘A’ for state 1 and ‘B’ for state 2. Given that we are using a copy of ‘**rd_simple1.inp**’, which denoted capture with a ‘1’, you should replace the ‘A’ with a ‘1’ (or start over and this time replace the ‘1’s’ in the input file with ‘0’ or some other code).

What code should you use for the second state? We are calling it state ‘U’ for ‘unobservable’, so you could use that code. However, it does not really matter, since by definition you never capture the animal while in that state.

Caution: Be sure, however, not to use zero as the code, because zero is *always* reserved to denote non-capture.

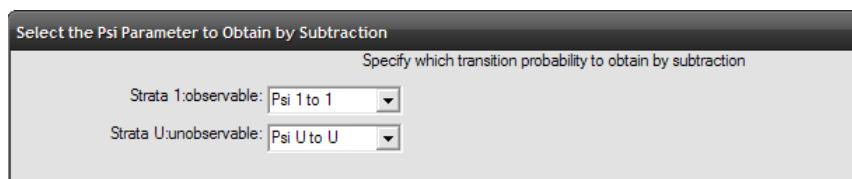
When this is complete, and after you return to the main screen and click ‘OK’, have a look at the PIM chart, shown below:



It is even more ‘busy’ than under the classic ‘**Robust Design**’ model. If animals in each state could be captured, then the PIM chart could remain this complex. However, in this case of an unobservable state we can, as it turns out, pare it down considerably. We will make three major changes in order to make this analysis identical to our previous analysis of these data: (1) change the PIM definitions, (2) equate survival for both states, and (3) fix the capture probabilities to 0 for the unobservable state.

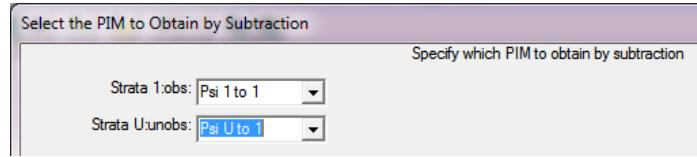
Changing the PIM definition would not be necessary for two of the three models we will run, but is necessary for one of them. It also permits us to equate γ values from the ‘**Robust Design**’ option with ψ values from the MSCRD model. Recall from Chapter 10 that under the multi-state model, transition probabilities for a given state need to sum to 1.0, thus one of the probabilities is computed by subtraction from 1.0. As with the ‘**Multi-state Recaptures only**’ option in MARK, the default is for the probability of remaining in a state to be gotten by subtraction. In the ‘**Robust Design**’ option in MARK the γ parameters always refer to being outside the study area, so we will mimic that case here using the ψ parameters in the MSCRD model.

First click on the ‘PIM’ menu, then on ‘**Change PIM definitions**’. This should spawn a window looking like the one shown below:



For each state you can designate which transition is obtained by subtraction. For state U change ‘Psi

U to U' to '**Psi U to 1'**. That way $\psi_i^{UU} = \gamma'_{i+1}$ will be estimated directly as a parameter.

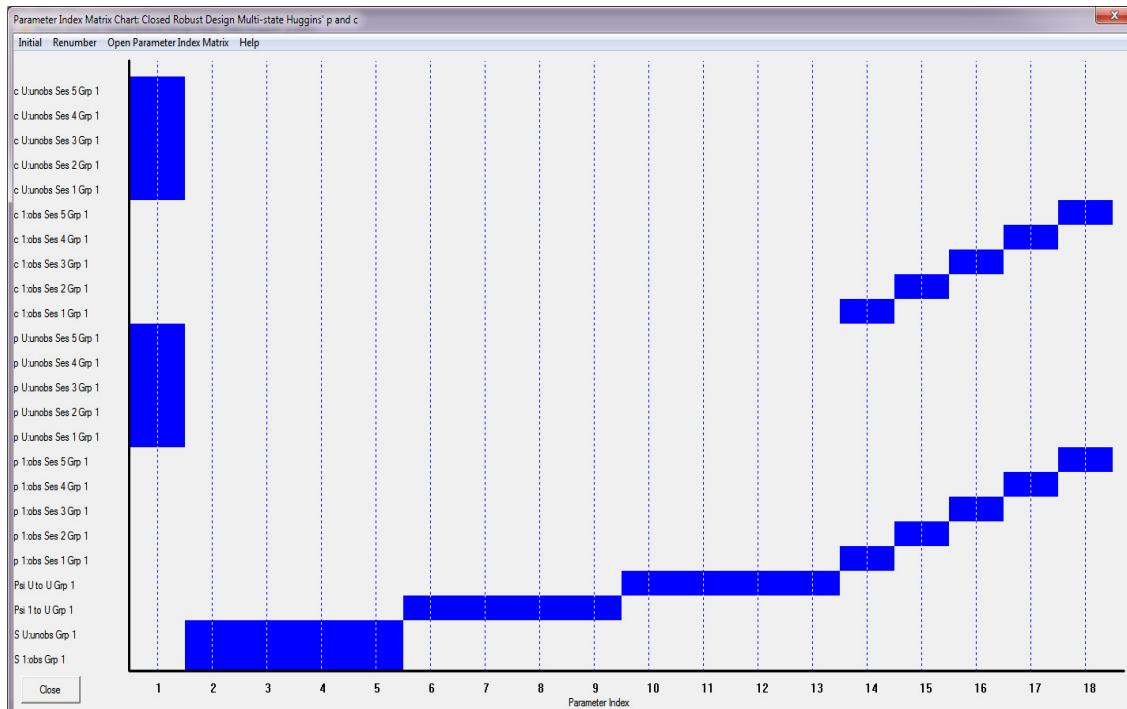


Next, you want to set survival equal for both states, which was one of the assumptions of the robust design model above. From previous chapters you will know that this can be done by dragging PIM's in the PIM chart, by copying one PIM to another after opening any PIM and then clicking on the '**Initial**' option at the top of the screen, or by opening a PIM and changing one entry at a time.

Finally, to account for the unobservable state you want to fix the capture and recapture probabilities for that state to 0. The simplest way to do it, especially when running many models, is to collapse all p and c PIM's for the unobservable state to one parameter. To make things easier, we'll designate that any parameter you are going to fix to 0 to be parameter number 1 (i.e., we'll move it to the far left of the PIM chart). In this way, you will always be fixing the same parameter to be 0. Otherwise, as you expand and contract the PIM chart with more restrictive or more general models, you will need to keep changing which parameter you fix to 0. Recall that parameters are fixed by going to the '**Run Current Model**' screen and clicking on '**Fix Parameters**'.

All of the encounter probabilities are set constant within primary period, and ' $c=p$ ' (corresponding to model M_o) – however, recall that the unobservable state '**U**' is just that – unobservable, and thus encounter probabilities is 0 for all primary occasions. Finally, under the assumptions of the robust design, we set the survival probabilities for observed and unobserved states equal to each other.

Here is the PIM chart – make sure you understand how it is set up:

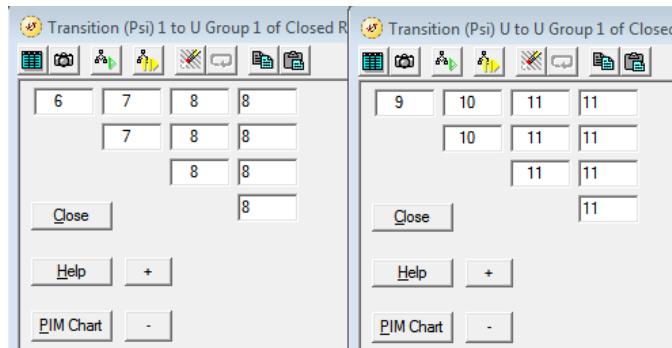


To compare the MSCRD approach against the usual robust design model we will set up and run the same three models that we used in section 16.6.1: ‘Markovian movement’, ‘Random movement’, and ‘No movement’. The following is a reminder of the constraints needed for each of the models, for both the classic ‘ γ ’ RD parametrization, and the equivalent MS closed RD.

	γ formulation	MS (ψ) formulation
no movement	$\gamma' = 1, \gamma'' = 0$	$\psi^{\text{UO}} = \psi^{\text{OU}} = 0$
random movement	$\gamma' = \gamma''$	$\psi^{\text{UU}} = \psi^{\text{OU}}$
Markovian movement	$\gamma'_k = \gamma'_{k-1}$ $\gamma''_k = \gamma''_{k-1}$	$\psi_k^{\text{OU}} = \psi_{k-1}^{\text{OU}}$ $\psi_k^{\text{UU}} = \psi_{k-1}^{\text{UU}}$

We’ll start with the ‘Markovian movement’ model. Because the last transition probabilities are confounded with the last survival parameter, set the last two equal to one another (i.e., $\psi_3^{\text{OU}} = \psi_2^{\text{OU}}, \psi_3^{\text{UU}} = \psi_2^{\text{UU}}$). This constraint is not necessary if (i) survival is set equal over time, or (ii) if ψ is modeled as a function of covariates. Recall that ψ_1^{UU} does not really enter the likelihood because no animals are released in the unobservable state. You can fix this parameter to any value (say, 0.5) or leave it alone. It will not affect the other parameters, but remember not to try to interpret it.

The PIMs for the movement parameters are shown below:



Go ahead and run this model – remember to fix parameter 1 (i.e., the encounter probability for the unobservable state) to 0 first. Here are the real parameter estimates for S and the ψ parameters:

Parameter	Estimate	Standard Error	95% Confidence Interval		Fixed
			Lower	Upper	
1:p U:unobs Session 1	0.0000000	0.0000000	0.0000000	0.0000000	
2:S 1:obs	0.6910107	0.0107492	0.6695553	0.7116729	
3:S 1:obs	0.7868343	0.0155368	0.7548019	0.8157037	
4:S 1:obs	0.9001978	0.0247887	0.8400541	0.9393585	
5:S 1:obs	0.9232715	0.0304891	0.8380912	0.9654840	
6:Psi 1 to U	0.1946989	0.0131690	0.1701755	0.2218115	
7:Psi 1 to U	0.3157277	0.0181494	0.2812780	0.3523283	
8:Psi 1 to U	0.2939751	0.0220538	0.2526711	0.3389685	
9:Psi U to U	0.5000000	0.0000000	0.5000000	0.5000000	
10:Psi U to U	0.1979315	0.0376099	0.1342830	0.2819241	
11:Psi U to U	0.3700734	0.0533519	0.2727986	0.4791784	

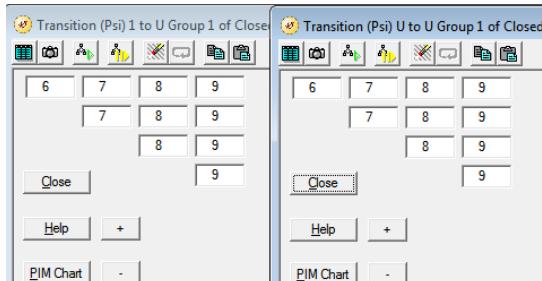
Note that they are virtually identical to the estimates for survival and γ from the Markovian model we fit using the classic closed RD.

It is also worth having a look at the estimated abundances \hat{N} :

Grp.	Sess.	N-hat	Standard Error	95% Confidence Interval	
				Lower	Upper
1	1	2985.0898	26.931161	2935.9953	3041.7877
1	2	1669.5593	12.583932	1647.6832	1697.2941
1	3	1147.4050	10.923456	1128.8150	1171.9726
1	4	1002.6467	16.336271	974.34657	1038.7721
1	5	921.43796	5.4395756	912.96067	934.71879
1	1	Not a Number	0.0000000	0.2225074E-307	0.2225074E-307
1	2	Not a Number	0.0000000	0.2225074E-307	0.2225074E-307
1	3	Not a Number	0.0000000	0.2225074E-307	0.2225074E-307
1	4	Not a Number	0.0000000	0.2225074E-307	0.2225074E-307
1	5	Not a Number	0.0000000	0.2225074E-307	0.2225074E-307

Since $M_{t+1} = 0$ for the unobservable state, then clearly we don't have a basis for estimating the size of the unobservable population, and therefore 'Not a number' is reported in this case.

For the '*Random movement*' model we do not need to constrain the last movement probabilities – movement is independent of which state you were in last time. So, we simply set $\psi^{OU} = \psi^{UU}$ (i.e., $\psi_1^{OU} = \psi_1^{UU}$, $\psi_2^{OU} = \psi_2^{UU}$, $\psi_3^{OU} = \psi_3^{UU}$...).



Go ahead and run this model – again fixing the encounter probability for the unobservable state to 0 first (note that for this model, this corresponds to parameter 1). Finally, recall that as was the case for the '*Markovian movement*' model, ψ_1^{UU} (parameter 6) does not enter the likelihood, but we constrain it for consistency. However, here we do not want to fix it to any value, since that would also imply fixing $\psi_1^{UU} = 0$, which we don't necessarily want to do.

Here are the results from fitting the MS '*Random movement*' model to the data – if you compare these estimates to those from the classic RD using ' γ ' notation, you'll see they are again virtually identical:

Parameter	Estimate	Standard Error	95% Confidence Interval		Fixed
			Lower	Upper	
1:p U:unobs Session 1	0.0000000	0.0000000	0.0000000	0.0000000	
2:S 1:obs	0.6954806	0.0107545	0.6740016	0.7161483	
3:S 1:obs	0.7742020	0.0135466	0.7465524	0.7996448	
4:S 1:obs	0.8784281	0.0184391	0.8374271	0.9101975	
5:S 1:obs	0.9338479	0.0000000	0.9338479	0.9338479	
6:Psi 1 to U	0.1998746	0.0131876	0.1752738	0.2269780	
7:Psi 1 to U	0.2877119	0.0151780	0.2589014	0.3183512	
8:Psi 1 to U	0.2918777	0.0202238	0.2538724	0.3330330	
9:Psi 1 to U	0.3019706	0.0000000	0.3019706	0.3019706	

Finally, for the 'No movement' model we fix $\psi_t^{OU} = \psi_t^{UU} = 0$, using the same PIM setup as the 'Random movement' model (above). Here is the results browser with all three models:

Model	AICc	Delta AICc	AICc Weight	Model Likelihood	No. Par.	Deviance	-2Log(L)
(MSCRD - Markovian movement model)	39171.3723	0.0000	0.96167	1.0000	14	39143.3400	39143.3400
(MSCRD - random model)	39177.8170	6.4447	0.03833	0.0399	12	39153.7930	39153.7930
(MSCRD - no movement model)	40028.9248	857.5525	0.00000	0.0000	9	40010.9110	40010.9110

Compare it against its counterpart from section 16.6.1, and you'll see they are identical. This drives home the point that the '**Robust Design**' and '**MSCRD**' options of **MARK** invoke two models that produce basically the same estimates. For the special case we have set up they represent the *exact* same likelihood, which reinforces the point that the '**Robust Design**' option represents a special case of the more general '**Closed Robust Design Multi-state**'.

In conclusion, if you can keep straight the definitions of the γ 's and their relationship with ψ 's, then for the case of one observable and one unobservable state, you can see from the examples we've shown that the '**Robust Design**' option is simpler to set up and deal with. The '**Closed Robust Design Multi-state**' option in **MARK** provides a more powerful and flexible tool for more complex scenarios that arise.

begin sidebar

the 'even flow' model

Back in section 16.3.4, we introduced an '*Even flow*' model. In the '*Even flow*' model, we are interested in whether the probability of moving from 'observable' at time (i) to 'unobservable' at time ($i + 1$) is the same as the probability of moving from 'unobservable' to 'observable' over the same interval. In other words, the '*Even flow*' model is specified by setting (i) $(1 - \gamma') = \gamma''$ in the classic closed RD, which is equivalent to (ii) setting $\psi^{OU} = \psi^{UU}$ in the multistate closed RD (MSCRD).

Let's consider the MSCRD formulation first. We'll used the simulated data set we analyzed before (`ms_rdsimple1.inp`; 5 primary periods). Recall from our earlier analysis of these data that we expect time-dependence in movement probabilities between observable and unobservable states. Thus, to construct the '*Even flow*' model, we set $\psi_1^{OU} = \psi_1^{UU}, \psi_2^{OU} = \psi_2^{UU}, \dots, \psi_4^{OU} = \psi_4^{UU}$. We should recognize at this point that we set $\psi_1^{OU} = \psi_1^{UU}$ for consistency only, since ψ_1^{OU} is undefined, and ψ_1^{OU} isn't in the likelihood. Thus, we would typically ignore the estimates over the first interval.

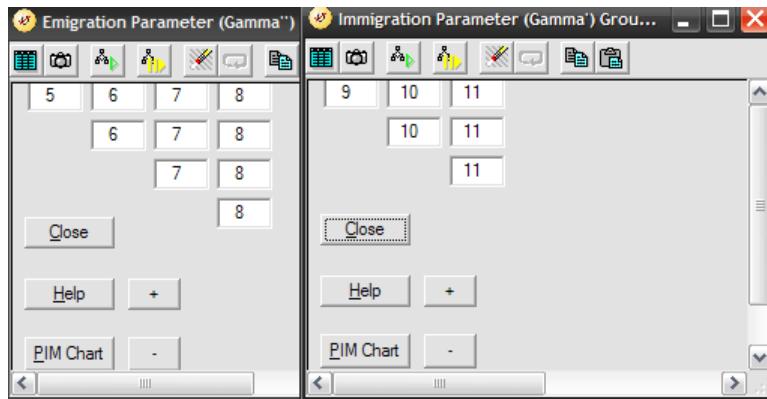
Bring up the results of your earlier MS analysis of these simulated data – retrieve the '*Random movement*' model. Next, we'll want to change the PIM definitions, to make sure we have the parameters ψ^{OU} and ψ^{UU} in model. Select the '**PIM | Change PIM definition**' menu, and make sure you specify '*Psi 1 to 1*' and '*Psi U to U*' as the transition probabilities to obtain by subtraction. Once you've done so, stop and think for a moment. Do you need to do anything more to construct the '*Even flow*' model? No – you're done! In the '*Random movement*' model we retrieved, we'd set $\psi_i^{UU} = \psi_i^{OU}$.

So, by simply changing the PIM definition so that ψ^{UU} is obtained by subtraction, then with the same PIM structure you retrieved from the '*Random movement*' model, but now with different definitions for those parameters, you've already 'built' the '*Even flow*' model. Go ahead and run this model, and add the results to the browser. You'll see that it doesn't do particularly well – better than the 'no movement' model, but much worse than the '*Random movement*' or '*Markovian movement*' models.

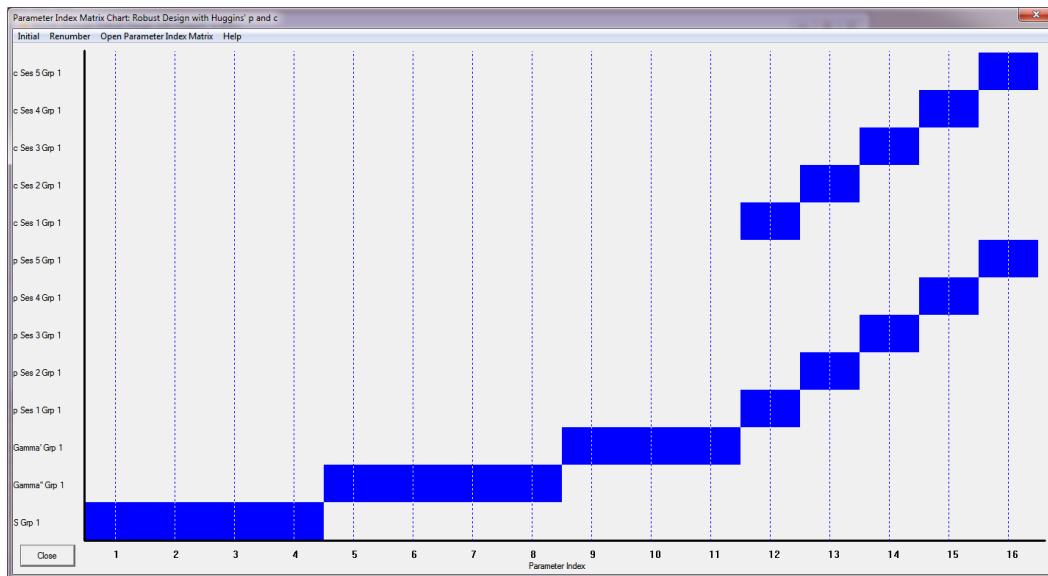
So far – pretty straightforward. Perhaps unexpectedly so for the MSCRD approach, given the convenience of switching directly from the '*Random movement*' to '*Even flow*' models simply by changing the PIM definition. However, what if instead we had used the classic ' γ ' parametrization of the closed RD? As noted earlier, fitting the '*Even flow*' model using γ notation means setting $(1 - \gamma') = \gamma''$.

OK – fine. But how do you set this equality constraint, when the model is parameterized using only γ' and γ'' , and where you are not able to specify that MARK used the complement of one of them? You can only ‘change PIM definitions’ for certain data types (such as the MS data type) – so how would set $(1 - \gamma') = \gamma''$? When we first introduced the ‘Even flow’ model back in 16.3.4, we made some cryptic mention of a ‘design matrix’ trick that you would need to use in order to construct the model using the ‘ γ parameterizations’. Time to introduce the ‘trick’.

First, go back and re-open your classic closed RD analysis of **rd_simple.inp**. Retrieve the ‘Markovian movement’ model in the browser. Open up the PIMs for γ' and γ'' , and eliminate the Markovian constraints (in other words, make them both fully time-dependent, with no overlapping parameter indices). The PIM structure for the two parameters should now look like:



Here is the corresponding PIM chart:



So, in effect, we’ve constructed a model with full time-dependence in S and both γ parameters.

For the next step, we want to build the DM corresponding to this ‘fully time-dependent’ model. At this point, this should be relatively straightforward for you.

One version of a DM corresponding to this PIM chart is shown below:

Design Matrix Specification (B = Beta)																
B1 S-int	B2 t1	B3 t2	B4 t3	B5 g-int	B6 gg	B7 t1	B8 t2	B9 t3	B10 gg.t2	B11 gg.t3	Parm	B12 p1	B13 p2	B14 p3	B15 p4	B16 p5
1	1	0	0	0	0	0	0	0	0	0	1:S	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	2:S	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	3:S	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	4:S	0	0	0	0	0
0	0	0	0	1	1	1	0	0	0	0	5:Gamma"	0	0	0	0	0
0	0	0	0	1	1	0	1	0	1	0	6:Gamma"	0	0	0	0	0
0	0	0	0	1	1	0	0	1	0	1	7:Gamma"	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0	0	8:Gamma"	0	0	0	0	0
0	0	0	0	1	0	0	1	0	0	0	9:Gamma'	0	0	0	0	0
0	0	0	0	1	0	0	0	1	0	0	10:Gamma'	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	11:Gamma'	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	12:p Session 1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	13:p Session 2	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	14:p Session 3	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	15:p Session 4	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	16:p Session 5	0	0	0	0	1

(Note: we've used an identity matrix structure for the S and p parameters, since we are not particularly interested in them here.)

We do want to pay attention to the DM modeling of the γ parameter, obviously. Here, we've simply adopted the familiar ‘intercept reference coding’ approach we've used much of the time so far in this book. Recall that what we're doing here is specifying β terms as relative deviances from a common ‘reference’ value (i.e., the intercept).

Alternatively, we could use an *identity* coding scheme (i.e., 1's along the diagonal) for the γ parameters:

5:Gamma"	1	1	0	0	0	0	0	0
6:Gamma"	1	0	1	0	0	1	0	0
7:Gamma"	1	0	0	1	0	0	1	0
8:Gamma"	1	0	0	0	1	0	0	0
9:Gamma'	0	0	1	0	0	0	0	1
10:Gamma'	0	0	0	1	0	0	0	0
11:Gamma'	0	0	0	0	1	0	0	0

Both would yield identical real estimates on the normal probability scale – the difference between the two is in terms of interpretation of the β parameters in the linear model. In either case, note the similarity of the structure of the part of the DM coding the γ parameters to a typical ‘age’ model (Chapter 7) – reflecting the fact that there is no γ' parameter for the first interval (such that the coding for time for γ' starts with interval 2).

Now – the ‘trick’. Not so much a ‘trick’, but rather a more advanced application of some DM concepts. We'll introduce the idea by first modifying our current time-dependent DM to specify the ‘Random movement’ model (where $\gamma'_i = \gamma''_i$). Recall from earlier in this chapter that to build the ‘Random movement’ model we applied a constraint to a DM with time-variation in the γ parameters (actually, we initially introduced it wrt to the ‘Markovian movement’ model, but the ‘Markovian movement’ model is essentially time-dependent with some constraints on the terminal pair of γ parameters). What we do here depends on which form of the DM we're using for γ . We'll proceed as if we're using the identity DM for γ – it's somewhat easier to explain (as you'll see).

So, for a ‘Random movement’ model, the DM corresponding to the γ parameters would look like

5:Gamma"	1	0	0	0
6:Gamma"	0	1	0	0
7:Gamma"	0	0	1	0
8:Gamma"	0	0	0	1
9:Gamma'	0	1	0	0
10:Gamma'	0	0	1	0
11:Gamma'	0	0	0	1

The modified DM if we’d used the intercept coding scheme would look like

5:Gamma"	1	1	0	0
6:Gamma"	1	0	1	0
7:Gamma"	1	0	0	1
8:Gamma"	1	0	0	0
9:Gamma'	1	0	1	0
10:Gamma'	1	0	0	1
11:Gamma'	1	0	0	0

Here we’ve structured the DM for the ‘Random movement’ model, which specifies that $\gamma'_i = \gamma''_i$. For the ‘Even flow’ model, we want to specify that $(1 - \gamma'_i) = \gamma''_i$. In other words, we want so set the complement of γ'_i equal to γ''_i . Our current DM for the ‘Random movement’ model is clearly pretty close, but how do we ‘tell it’ to use the complement of γ'_i ?

In fact, it turns out that you can specify the equality of one parameter with the *complement* of another using a ‘1, -1’ coding scheme, where we use the ‘1’ to indicate one parameter, and ‘-1’ to indicate the complement of the other. In this case, we would use ‘1’ to code for $(1 - \gamma'_i)$, and ‘-1’ to code for γ''_i . Using the identity DM, we simply need to change the dummy coding for each time step using the ‘1, -1’ coding convention (as described).

Here is the completed DM for the γ parameters, corresponding to the ‘Even flow’ model:

5:Gamma"	1	0	0	0
6:Gamma"	0	-1	0	0
7:Gamma"	0	0	-1	0
8:Gamma"	0	0	0	-1
9:Gamma'	0	1	0	0
10:Gamma'	0	0	1	0
11:Gamma'	0	0	0	1

If you run this model, you’ll see that it gives you exactly the same value for $-2 \log(\mathcal{L})$ as we obtained for the ‘Even flow’ model using the MSCRD approach (above). What if instead of $(1 - \gamma'_i) = \gamma''_i$ we wanted $(1 - \gamma''_i) = \gamma'_i$ (which would correspond to setting $\psi_i^{\text{UU}} = \psi_i^{\text{UO}}$)? Easy – simply reverse the DM coding so that ‘-1’ is used for γ'_i , and ‘1’ is used for γ''_i .

Now, as a real test of your understanding, how would we modify the intercept-based DM for a ‘Even flow’ model? The trick is to think – hard – about what the intercept represents. The ‘answer’ is shown at the top of the next page – no peeking!

See if you can figure it out on your own. It’s somewhat trickier than the identity matrix approach we demonstrated first, but if you understand what this modified DM shows (i.e., if you understand the ‘strange’ things we did to the intercept) then congratulations are in order, since that would exhibit a pretty solid understanding of the DM, and intercept coding in general.

5:Gamma"	-1	-1	0	0
6:Gamma"	-1	0	-1	0
7:Gamma"	-1	0	0	-1
8:Gamma"	-1	0	0	0
9:Gamma'	1	0	1	0
10:Gamma'	1	0	0	1
11:Gamma'	1	0	0	0

Nifty stuff, eh? Bonus points if you can figure out why this ‘trick’ works (i.e., what you’re really doing with ‘1’ and ‘-1’ coding). Actually, if you understand the linear model being constructed, it isn’t too bad.

end sidebar

16.8. The ‘open’ robust design...

Our discussion here of the robust design assumes that the closure assumption within a primary period is valid. In Chapter 15 we outlined conditions, discussed in Kendall (1999), where certain types of violation of closure do not induce bias in estimators. These same conditions are directly applicable here as well. However, one situation where this will not work well is where both arrivals and departures from the study area are occurring throughout the primary period. This situation falls under the umbrella of an ‘open robust design’, which we describe here.

16.8.1. Background

The ‘Open Robust Design multi-state’ data type in MARK (hereafter, MSORD) derives from Kendall & Bjorkland (2001) and Kendall *et al.* (2019), based on the design first described by Schwarz & Stobo (1997). We’ll use the case of nesting sea turtles from Kendall & Bjorkland (2001) to motivate the use of this data type, as well as how to use it. Schwarz & Stobo (1997) used the case of a rookery of grey seals, and we also believe it could be quite useful for stopover studies of migratory species.

In a study of sea turtles there is an interest in estimating survival probabilities, breeding probabilities, and perhaps population size (as well as population growth rate). Nesting seasons are extensive, up to five months. Capture effort is typically throughout the season, in some cases nightly. Because sea turtles often lay more than one clutch, there is an opportunity to recapture a given female multiple times in a season. In summary, sampling for a given year consists of multiple sampling periods, where each individual in the nesting population has a chance (assumed to be the same chance) of being captured in each sampling interval. With a couple of additional assumptions, this constitutes a robust design.

In the preceding sections of this chapter, we described the closed robust design, where it was assumed that, for the duration of capture effort within a primary period, one of the following was true: (1) the population occupying the study area was completely closed to additions or deletions, (2) individuals moved completely randomly in and out of the study area, (3) all individuals were present in the first sampling occasion within a primary period, although marked and unmarked individuals could exit the study area (with the same probability) before the last sampling occasion for that season, or (4) individuals could enter the study area between the first and last sampling occasion within a season, assuming all individuals are present by the last sampling occasion. An additional assumption for conditions 3 and 4 is that capture probability within a primary period varies only by time (not trap response or individual heterogeneity).

In the case of nesting sea turtles, or marine mammal rookeries, the above assumptions do not hold. In

fact, turtles arrive to lay their first clutch in a staggered fashion, remain in the area to renest for variable periods of time, then complete nesting and return to foraging areas in a staggered fashion. In essence, there is an open population study going on within each nesting season. First arrival at the nesting beach is equivalent to birth, and departure for the foraging grounds after the last clutch is laid is equivalent to death in a modeling sense.

If each individual in the population could be relied upon to be on the nesting beach each year, then the data for the entire nesting season could be pooled into whether or not an individual was captured in year (t). However, some individuals skip nesting in a given year, and therefore the nesting population and population of female breeders in a given year are not equivalent. If nesting were a completely random process (i.e., each adult female had the same probability of nesting), then a CJS analysis from pooled data would produce an unbiased estimator for survival, although breeding probability could not be estimated. With most species, however, breeding probability is more accurately characterized as a Markov process (i.e., the probability of breeding is dependent on whether or not an individual is currently a breeder), and for some species skipping at least one year after breeding is obligatory. In this case, if skipping is not accounted for, all estimators in the CJS model, including survival, will be biased.

16.8.2. The General ORD Model

The essence of any robust design model is to take advantage of multiple sampling periods over a sufficiently short period of time that the state of the individual (e.g., nester or non-nester) remains static, in order to estimate the effective capture probability for those that are observable in that primary period (e.g., nesters). Because of the possibility of some individuals occupying an unobservable state (i.e., away from the study area[s]), we use a multistate approach to model the capture process across primary periods. For modeling captures within a primary period we use a generalization of the Schwarz-Arnason (1996, Biometrics) version of the Jolly-Seber model (see Chapter 12). The details of this generalization will become apparent below, but basically the probability an animal remains in the study area from one sampling period to the next can be modeled as a function of time (as in the Schwarz-Arnason model) or the number of sampling periods since it first arrived that season (i.e., its 'age' within the season).

The ORD model first conditions on the total number of individuals captured in primary period (t). Given that an individual is captured at least once within primary period (t), the model then considers the probability of each observed capture history within that primary period. For example, if a nesting turtle is first captured on capture occasion 2 (of 6) of year (t), the model considers two possibilities. She could have arrived to lay her first clutch during the first capture occasion (with probability $pent_{t_1}$), was not captured on that occasion (with probability $1 - p_{t_1}$), then returned to nest again (with probability $\varphi_{t_{1,0}}$) where time subscript 1 indicates sampling period 1 and age subscript 0 implies this is the first clutch laid this season by this female) and was captured (with probability p_{t_2}) during capture occasion 2. Alternatively, she could have arrived to lay her first clutch during capture occasion 2 (with probability $pent_{t_2}$). So for six capture occasions within a year (i.e, one primary period), we have the following probability structure for the history 010111:

$$\left[pent_{t_1} (1 - p_{t_1}) \varphi_{t_{1,0}} \varphi_{t_{2,1}} \varphi_{t_{3,2}} \varphi_{t_{4,3}} \varphi_{t_{5,4}} + pent_{t_2} \varphi_{t_{2,0}} \varphi_{t_{3,1}} \varphi_{t_{4,2}} \varphi_{t_{5,3}} \right] p_{t_2} (1 - p_{t_3}) p_{t_4} p_{t_5} p_{t_6},$$

which can be rewritten as

$$\begin{aligned} & pent_{t_1} (1 - p_{t_1}) \varphi_{t_{1,0}} p_{t_2} \varphi_{t_{2,1}} (1 - p_{t_3}) \varphi_{t_{3,2}} p_{t_4} \varphi_{t_{4,3}} p_{t_5} \varphi_{t_{5,4}} p_{t_6} \\ & + pent_{t_2} p_{t_2} \varphi_{t_{2,0}} (1 - p_{t_3}) \varphi_{t_{3,1}} p_{t_4} \varphi_{t_{4,2}} p_{t_5} \varphi_{t_{5,3}} p_{t_6}. \end{aligned}$$

Because a turtle only arrives to lay her first clutch once, the entry probabilities ($pent_{t_i}$) have to add

to 1.0. Once captured within a year, subsequent captures within that year are modeled as a function of future ‘survival’ (in this case the probability a turtle keeps coming back to lay more clutches) and capture probability.

In summary, the following parameters will be found in the MSORD data type in **MARK**: S_t^r = survival from primary period (t) to ($t + 1$) for those occupying state r during period (t); ψ_t^{rs} = probability an individual in state r in primary period (t) is in state s in primary period ($t + 1$), given it survives to period ($t + 1$); p_{tj}^s = probability that an individual in state s in primary period (t) is a new arrival (within that primary period) to the study area for that state at capture occasion j ; $\varphi_{tj,a}^s$ = probability that an individual in the study area associated with state s at capture occasion (j), and who first arrived in the study area a capture occasions previous, is still in that study area at capture occasion ($j + 1$); and p_{tj}^s = probability that an individual in the study area for state s at capture occasion (t_j) is captured. Of course any of these parameters can also be group- (e.g., sex) or true age-dependent.

Although useful and powerful, the use of the ORD in combination with MS models at least initially raises the dimensionality of the problem of programming models in **MARK**. As with the MSCRD model introduced in section 16.7, there are PIM’s for state-specific survival between primary periods, and for state-specific transitions between primary periods. For each primary period there is a PIM for p_{ent} , φ , and p for each group and state, whereas for the MSCRD there were PIM’s for p and c . The ORD also raises the dimensionality of model selection, where you explore variation in parameters both across primary periods (S , ψ , p_{ent} , φ , and p) and within primary periods (p_{ent} , φ , and p).

16.8.3. Implementing the ORD in MARK: (relatively) simple example

To illustrate some the points made above we will continue the example of nesting sea turtles on a single beach. We will consider a five-year study. Data are collected nightly on the beach in question, for three months. When laying multiple clutches, females space those clutches approximately two weeks apart. In dividing the season into capture occasions, it makes sense to do it so that each time an individual re-nests she has a chance of being included in a capture occasion. Therefore we divide the three-month season into six half-month capture occasions (if an individual is captured one or more times in within the half-month interval you record a ‘1’ in the capture history).

An example history for a five-year study, each with six capture occasions (totaling $(5 \times 6) = 30$ capture occasions for the study) is

| 00101000000001111000000001001 1;

In this case a female is captured in sampling periods 3 and 5 in year 1, sampling periods 3-6 in year 3, and sampling periods 3 and 6 in year 5. As with other models in **MARK**, you provide the total number of encounter occasions (in this case 30). As with the closed robust design, when you designate the MSORD option in **MARK**, it provides a screen titled ‘**Easy Robust Times**’, which is an aid to specifying how the capture history should be broken up into primary periods and capture occasions. **MARK** will ask how many primary occasions there are (in this case 5). **MARK** will then provide a screen indicating equal numbers of capture occasions per primary period. However, the MSORD model does not require them to be equal, and **MARK** allows you to correct these values.

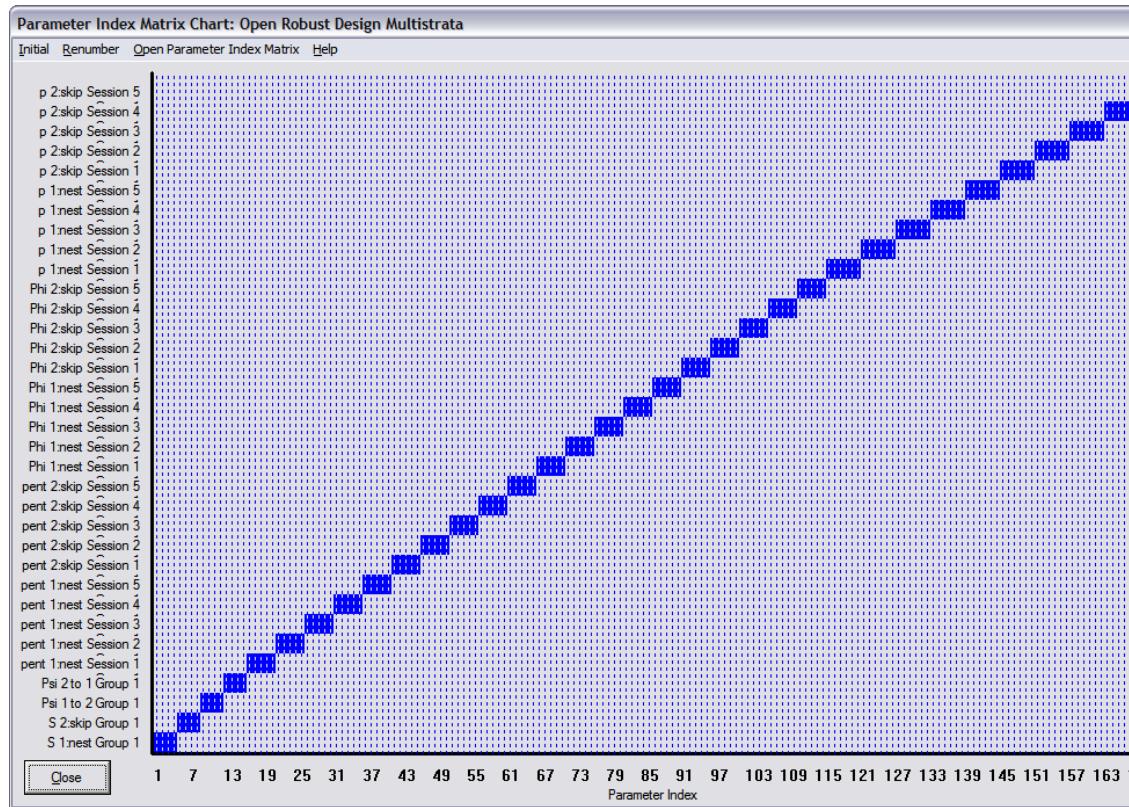
As with the ‘**Multi-state Recaptures only**’ (Chapter 10) feature of **MARK**, after specifying the number of states you go to the ‘**Enter State Names**’ screen, where you designate the label (the code used in the encounter history to designate the state), and name for each state. In the case of the example capture history above, where we have a single-site study, with one unobservable state, we would replace the **MARK** default of ‘A’ with a ‘1’ in the label (to be consistent with the encounter history shown above), and might name it ‘nest’ (meaning that the animal was observed nesting). We can name the other state

with something like 'skip' (because the animal was skipping nesting).



For this unobservable state it does not matter what label you give it (but do not make it '0' or the same as state 1), because animals are neither released nor observed in that state anyway.

When you have completed the model specification screen **MARK** will set up the PIM's for the '**MSORD**' model. Before we look at individuals PIMs, it's worth firing up the PIM chart, if for no other reason than to 'impress yourself' (and perhaps help convince your employers you need a bigger monitor). As noted earlier, PIM charts for robust design models can be 'busy' – the default time-dependent PIM chart for the turtle data is shown below.



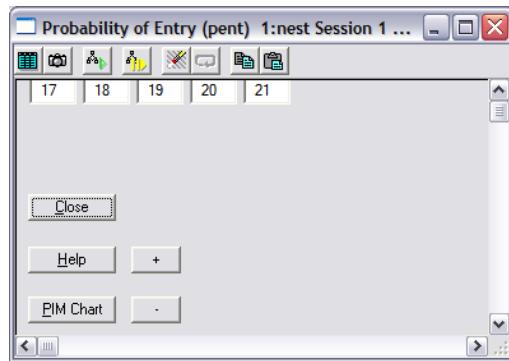
Pretty scary – it's so dense with information, you can barely read some of the labels on the left-hand side. As such, we'll do much of our manipulation of the basic parameter structure for our models using the individual PIMs.

Because this is a multi-state model, the PIM's for S and ψ are structured just as with the MS model (discussed at length in Chapter 10). They are upper triangular matrices, where you can specify these

parameters to be constant, time-specific, partially age-specific, release-cohort specific, etc. You can also apply specific constraints to ensure transitions sum to 1. You can even specify which transitions are reported by MARK.

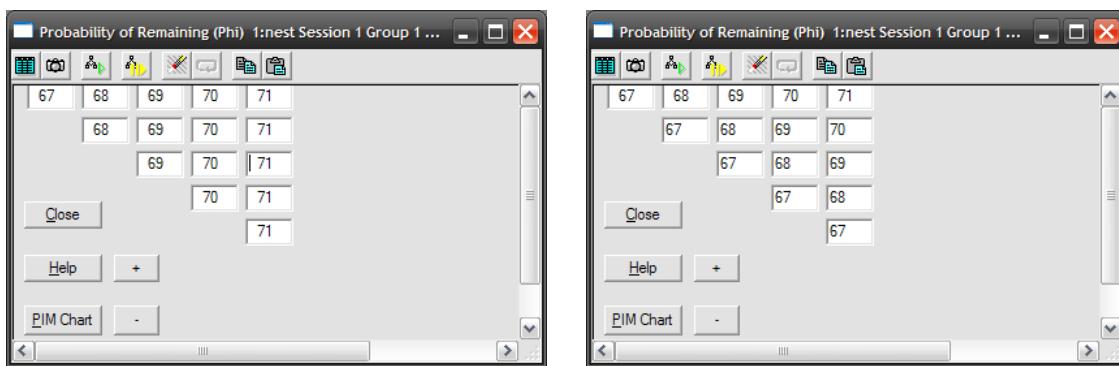
For parameters relating to capture histories within primary period, the PIM's for p_{ent} and p are really vectors, implying they can be modeled as a function of time or covariates, but not time since capture within the primary period.

For example, here is the PIM for p_{ent,t_1} :



The PIM's for φ are of the typical format (i.e., an upper triangular matrix). However, keep in mind that typically the rows of a PIM denote a capture cohort, thereby permitting a parameter to be modeled as a function of time since first capture. For the φ PIM's the rows denote an *entry* cohort, permitting one to model these parameters as a function of the time since arrival to the study area (e.g., for nesting turtles that probability a female lays another clutch is a function of the number of clutches she has laid so far).

We provide two examples of PIM's below.

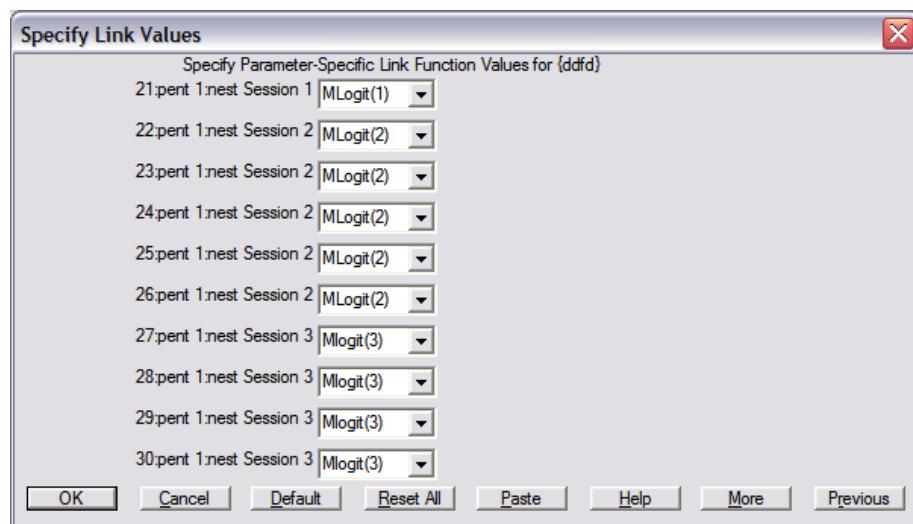


Whereas in the first (left-most) example φ is modeled strictly as a function of capture occasion, in the second example (right-most) it is modeled strictly as a function of the number of capture occasions since first arrival.

In the first PIM, parameter 68 refers to the probability that an animal in the study area at capture occasion 2 will still be in the study area at capture occasion 3, independent of whether the animal was present or not present on occasion 1. In the second PIM above parameter 68 refers to the probability that an animal in the study area at any capture occasion (j) will still be in the study area at capture occasion ($j + 1$), given that the animal has been in the study area for two capture occasions, implying it

first entered the study area that season at capture occasion ($j - 1$) (e.g., with sea turtles the individual laid her second clutch at capture occasion j).

There is an important point to consider about $pent$, the probability an animal arrives at the study area before any given sampling period, given that it arrives at some time during the season. For a given primary period the probability of entry across all sampling periods must add to 1.0 (i.e., $\sum_{j=1}^{l_t} pent_j = 1.0$). **MARK** derives the final (terminal) $pent_{l_t}$ by subtraction, and *therefore you cannot model this parameter directly*. In order to satisfy this constraint reliably you should use the multinomial logit (**mlogit**) link in **MARK**, just as with the multistate models as described in Chapter 10. This is invoked in the 'Run' screen by specifying the '**parm-specific**' option for the link function.



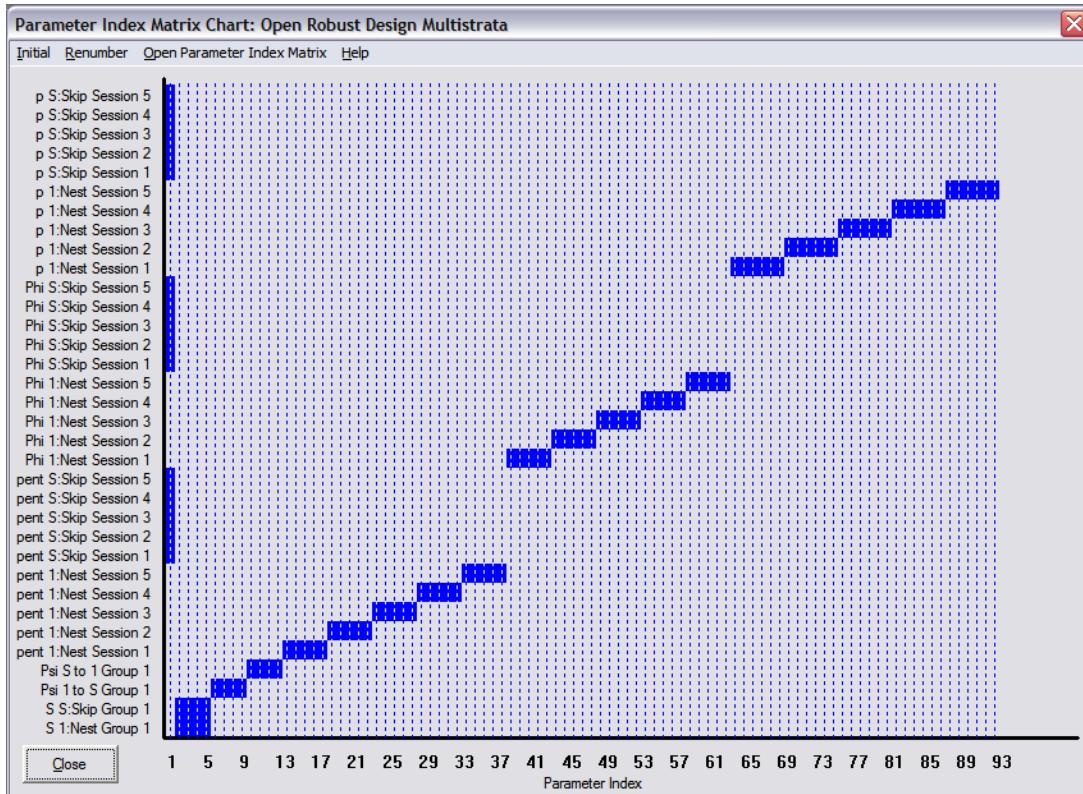
Each series of parameters that must add to one gets the same **mlogit** designation. For example, all $pent$'s for primary period 1 in state 1 would be assigned **mlogit(1)**, all $pent$'s for primary period 2 in state 1 would be assigned the **mlogit(2)** link, and so on. If you fail to do this you will most likely get an error message saying the numerical convergence was not reached. The entry probabilities are especially prone to this type of problem, because there are potentially so many different estimates that must sum to 1.0.

16.8.4. Dealing with unobservable states

Accounting for unobservable states with the '**MSORD**' feature of **MARK** is different than doing so in the original Robust Design option in **MARK** (discussed earlier in this chapter). With the latter, the model is set up explicitly for the case of one study area plus temporary emigrants. The fact that temporary emigrants actually occupy an unobservable state is treated implicitly. That model includes one PIM for survival (assumed the same for those in the study area and those outside), two PIM's for the temporary emigration process (coming and going), and one PIM for each primary period for detection probabilities in the study area. Conversely, the ORD model is nested within a general MS model framework. Therefore there will be PIM's for the within-primary-period parameters ($pent$, φ , and p) for each state. For a T -primary period study involving V states and G groups, this implies there will be $G(V \times 2 + V \times T \times 3)$ PIM's.

This framework is very flexible for dealing with unobservable states, because *an unobservable state*

is simply one where capture probability is always 0. However, because of this flexibility there is also some irritation involved with dealing with all those PIM's, many of which do not get used. The PIM chart shown below illustrates a model from our example of one adult female population of sea turtles, where there is one observable (nesting) and one unobservable (skipping) state.



First, this chart illustrates yet again how this model can quickly make working directly with the PIM chart impractical (there are 34 PIM's in this relatively simple case of 5 years, 2 states, and 1 group). Second, it illustrates how within-primary-period parameters for the unobservable state are dealt with. Here we have set the $pent$'s, φ 's, and p 's for the unobservable state equal to 0, after assigning all these parameters to parameter index 1 (because these parameter will be set to 0 for each model considered, assigning them index 1 prevents you from being required to fix a different parameter to 0 for each run). Fixing the p 's for the unobservable state to 0 is most important, because this implies the effective capture probability for the primary period will be 0. Once you do that, it does not matter what value you give to the $pent$'s or φ 's because they never enter into the model as the animal is 'uncatchable' (i.e., unobservable in this state).

Also, note that we have set the survival PIM for the skipped breeders equal to that for the nesters. This is done implicitly in the original robust design model, but is necessary to do explicitly here as well. This constraint makes sense, since one cannot directly monitor survival of the unobservable animals, because they cannot be captured and released in the unobservable state. In general, it is a price to be paid for the fact that an unobservable state creates missing cells of data. However, under the assumption that survival is the same for both states, there is enough indirect information (from marked animals leaving and coming back) to estimate the transition probabilities ψ .

16.8.5. Which parameters can be estimated?

Identifying which parameters can be estimated can be a tricky business with these models, as it is for the other models in **MARK**. The first question is which parameters can be estimated based on the structure of the model (assuming no sparseness in the data). This issue is discussed for a single observable state in Kendall & Pollock (1982), Kendall *et al.* (1997), Kendall & Bjorkland (2001), Kendall & Nichols (2002), Fujiwara & Caswell (2002), and Schaub *et al.* (2004).

- If there are no unobservable states, then under the ORD all S 's and ψ 's, for each time period and state, can be estimated (i.e., since the effective capture probability is estimable from the within-primary-period data, there is no confounding of parameters in the last period).
- For the case of one observable and one unobservable state, under a robust design, if $\psi_{T-1}^{(obs \rightarrow unobs)}$ and $\psi_{T-1}^{(unobs \rightarrow obs)}$ can be constrained to equal their counterparts for an earlier time period, then all the other parameters can be estimated as time specific.
- For multiple observable or unobservable states, investigations into estimability are ongoing, and call for methods such as described in Gimenez *et al.* (2004), and alternative numerical methods (see Appendix F) to investigate which parameters or combinations of parameters can be estimated, given the structure of the model. There are also some parameters within a primary period that cannot be estimated under the most general models. For the case where $pent$, φ , and p are all time dependent, $pent_{t_1}$, p_{t_1} , and $pent_{t_2}$ are all confounded, as are $\varphi_{t_{l-1}}$, p_{t_1} , and $pent_{t_l}$ (Kendall & Bjorkland 2001).

16.8.6. Goodness of fit

As with other CMR models, there is no perfect answer to the question of how to assess absolute fit of the MSORD model to your data set. The only test specific to this model, for the case of one observable and one unobservable state, is a Pearson χ^2 test (pooling cells with small expected frequencies) available in program **ORDSURVIV** (Kendall & Bjorkland 2001, www.pwrc.usgs.gov/software).

16.9. Derived parameters based on information contained within primary periods

In addition to the parameters listed above that are part of the '**MSORD**' model **MARK** also reports estimates of five other derived parameters for each state and primary time period:

1. the number of animals in that state in that primary period, $\hat{N}_t^{*,s}$;
2. the residence time (or stopover time), \hat{R}_t^s , the average number of secondary sampling periods that an individual in state (s) spent in the study area in primary period (t);
3. the intensity parameter, $\alpha_{t_j}^s$, the probability in individual in state s is present in the study area in secondary sampling period j of primary period (i);
4. expected arrival period, $a_{t_j}^s$, the average secondary sample the individual in state s is first available for capture;
5. expected departure period, $g_{t_j}^s$, the average secondary sampling period that an individual in state s is available for capture.

Derived parameters: expected residence time: We include two approaches to estimating expected residence time, R_{it}^s , for observable state (s) and primary period (t): the number of sampling periods between the arrival to and departure from the study area of individual (i) .

Expected residence time under Method 1 ($R_t^s(1)$) assumes that the field season begins no later than the arrival of the first individual, and concludes no earlier than the departure of the last individual. In other words, the time period during which the population uses the study area is bounded by sampling effort. In this case $R_t^s(1)$ is computed as a weighted average, where the weights are the probabilities that an individual remains for 1, 2, 3, ... time periods. For three secondary samples:

$$\begin{aligned} E(R_t^s) = & 1 \times [pent_{t_1}^s (1 - \varphi_{t_1,0}^s) + pent_{t_2}^s (1 - \varphi_{t_2,0}^s) + pent_{t_3}^s] \\ & + 2 \times [pent_{t_1}^s \varphi_{t_1,0}^s (1 - \varphi_{t_2,1}^s) + pent_{t_2}^s \varphi_{t_2,0}^s] \\ & + 3 \times pent_{t_1}^s \varphi_{t_1,0}^s \varphi_{t_2,1}^s. \end{aligned}$$

Expected residence time under Method 2 ($R_t^s(2)$) assumes that persistence in the study area in a given primary period is a function only of the number of sampling periods since arrival. For three secondary samples:

$$\begin{aligned} & 1 \times (1 - \varphi_{.,0}^2) + \\ & 2 \times (\varphi_{.,0}^s) \times (1 - \varphi_{.,1}^s) + \\ & 3 \times (\varphi_{.,0}^s) \times \varphi_{.,1}^s. \end{aligned}$$

The subscripts indicate persistence varies only by number of periods since first arrival. In this case it is still assumed that sampling effort begins no later than when the first individual arrives at the study area. It does not require that sampling effort continue until the last individual leaves, but does assume that the probability individual (i) remains for the entire field season is small (i.e., so that $\varphi_{.,v}$ can be estimated for all pertinent values of v).

Derived parameters: phenology parameters intensity of availability, expected arrival period, and expected departure period: Intensity of availability, $\alpha_{t,j}^s$, is the probability, given an individual is in state s in primary period t , that the individual is present and available for detection in secondary sampling period j . For example, the probability an individual in state s is present in secondary period 3 is

$$\alpha_{t_3}^s = pent_{t_1}^s \varphi_{t_1,0}^s \varphi_{t_1,1}^s + pent_{t_2}^s \varphi_{t_2,0}^s + pent_{t_3}^s.$$

Expected arrival period for an individual in state s is a weighted average of the period-specific arrival probabilities. For three secondary samples:

$$a^s = (1 \times pent_1^s) + (2 \times pent_2^s) + (3 \times pent_3^s).$$

Expected departure period (departing immediately after the period) is also a weighted average.

For the example of three secondary periods:

$$\begin{aligned} g^s = & 1 \times pent_{t_1}^s (1 - \varphi_{t_1,0}^s) \\ & + 2 \times [pent_{t_1}^s \varphi_{t_1,0}^s (1 - \varphi_{t_2,1}^s) + pent_{t_2}^s (1 - \varphi_{t_2,0}^s)] \\ & + 3 \times [pent_{t_1}^s \varphi_{t_1,0}^s + pent_{t_2}^s \varphi_{t_2,0}^s + pent_{t_3}^s]. \end{aligned}$$

The calculation of expected arrival and departure probabilities is conditional on the same assumption as Method 1 for estimating expected residence time: that the span of sampling effort at least matches use by the population (i.e., it could be longer, but not shorter).

16.10. Alternative parameterization of multistate robust design models to estimate ‘state structure’

For both the **MSCRD** and **MSORD** models, there is an alternative parameterization that includes for each primary period the set of parameters $\{\omega_t^s\}$ representing state structure, the probability that an individual is in each state during primary period (t). We chose to use ω to match the use in the matrix modeling book by Caswell (2001).

We can define the parameter, using the example of two states **A** and **B**:

$$\omega_t^A = \frac{N_t^A}{N_t^A + N_t^B}, \quad \text{and} \quad \frac{N_t^B}{N_t^A + N_t^B} = 1 - \omega_t^A.$$

From this you’ve probably figured out that you could derive this parameter from the derived abundance parameters that are produced by the models as already discussed. But (1) you would have to use the Delta method or MCMC (Appendix B) to derive the variances for these estimators, and (2) you might be interested in modeling ω directly as a function of covariates, a trend, etc., and therefore there is extra value in it being a parameter of the model. For example, if states **A** and **B** are breeder and non-breeder, ω could be used to address hypotheses about what drives breeding propensity.

The only difference in model structure for this version of the **MSCRD** and **MSORD** models is within each primary period. For the default parameterizations of these models, for primary period (t), consisting of three secondary samples, where detection probability varies only by time (no trap response, no individual heterogeneity), the model structure for the capture history ‘B0B’ would be

$$\frac{p_{t_1}^B(1-p_{t_2}^B)p_{t_3}^B}{p_t^{*B}} \quad \text{or} \quad \frac{pent_{t_0}^B p_{t_1}^B \varphi_{t_1}^B (1-p_{t_2}^B) \varphi_{t_2}^B p_{t_3}^B}{p_t^{*B}}$$

for the **MSCRD** or **MSORD** models, respectively.

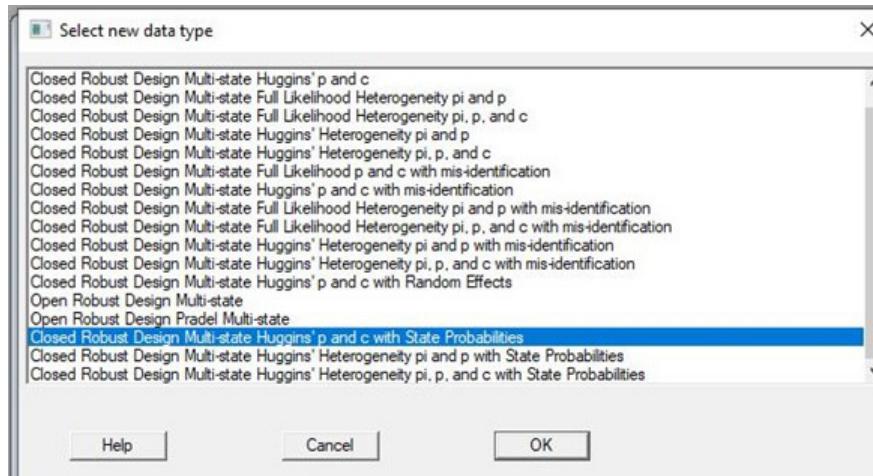
For these same models with the ω parameterization, the structure for history ‘B0B’ would be

$$\frac{\omega_t^B p_{t_1}^B (1-p_{t_2}^B) p_{t_3}^B}{\omega_t^A p_t^{*A} + \omega_t^B p_t^{*B}} \quad \text{or} \quad \frac{\omega_t^B pent_{t_0}^B p_{t_1}^B \varphi_{t_1}^B (1-p_{t_2}^B) \varphi_{t_2}^B p_{t_3}^B}{\omega_t^A p_t^{*A} + \omega_t^B p_t^{*B}}$$

for the **MSCRD** or **MSORD** models, respectively.

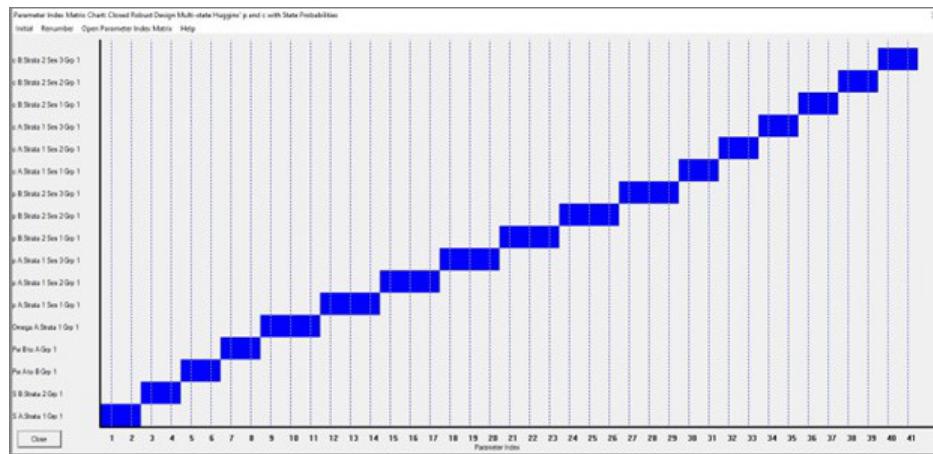
16.10.1. Implementing the ω parameterization

The ω parameterization can be accessed from within the **MSCRD** and **MSORD** model structures, using the ‘PIM | Change Data Type’ option in **MARK**. For the **MSCRD** model, you be presented with the windows shown at the top of the next page.



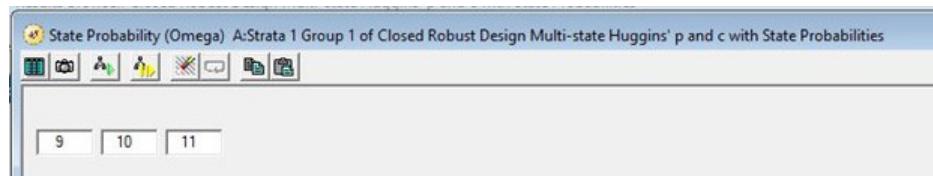
and choose the '**Closed Robust Design Multi-state ... with State Probabilities**' option.

Have a look at the PIM chart for this data type:



and you will see that there is now a PIM for ω between the PIMs for ψ and the PIMs for p .

Looking at this PIM, for the case of three primary periods and two states **A** and **B**,



you see that you get the PIM for ω^A . Because the ω 's have to add to 1.0 across states, you will get a PIM for ω for each state except the last listed when you set up the model (in this case state **B**), which is derived by subtraction. This is another parameter type where, for more than two states, it's advisable to use the `mlogit` link function (introduced in Chapter 10), to ensure estimates don't sum to > 1.0.

We've included output from this model for an example data set with 3 primary periods

MSCRD Example analysis classic param				
Real Function Parameters of {global corr with omega} corr				
Parameter	Estimate	Standard Error	95% Confidence Interval	
1:S A:Strata 1	0.7042490	0.0308602	0.6404251	0.7609724
2:S A:Strata 1	0.7668568	0.0373395	0.6859610	0.8320171
3:S B:Strata 2	0.7920432	0.0279475	0.7319856	0.8415559
4:S B:Strata 2	0.7938889	0.0360429	0.7143888	0.8557290
5:Psi A to B	0.1600519	0.0269486	0.1139777	0.2201237
6:Psi A to B	0.2306715	0.0270579	0.1819238	0.2878847
7:Psi B to A	0.2068416	0.0286031	0.1563302	0.2684800
8:Psi B to A	0.2394873	0.0269129	0.1907551	0.2961137
9:Omega A:Strata 1	0.5116703	0.0226737	0.4672546	0.5559024
10:Omega A:Strata 1	0.5158162	0.0199458	0.4767054	0.5547343
11:Omega A:Strata 1	0.5029152	0.0192366	0.4652662	0.5409312
12:p A:Strata 1 Session	0.3081211	0.0253577	0.2607533	0.3599044
13:p A:Strata 1 Session	0.3159713	0.0256982	0.2678826	0.3683493
14:p A:Strata 1 Session	0.3591475	0.0275118	0.3071770	0.4146484
15:p A:Strata 1 Session	0.3146883	0.0217759	0.2736466	0.3588444
16:p A:Strata 1 Session	0.3296735	0.0222326	0.2876453	0.3746131
17:p A:Strata 1 Session	0.3379986	0.0224801	0.2954329	0.3833601
18:p A:Strata 1 Session	0.3457591	0.0230092	0.3021465	0.3921295
19:p A:Strata 1 Session	0.3784660	0.0240533	0.3325923	0.4266225
20:p A:Strata 1 Session	0.3332993	0.0225978	0.2905688	0.3789570
21:p B:Strata 2 Session	0.3557501	0.0265589	0.3055548	0.4093308
22:p B:Strata 2 Session	0.3434119	0.0261111	0.2941946	0.3962405
23:p B:Strata 2 Session	0.3763137	0.0272864	0.3245161	0.4311023
24:p B:Strata 2 Session	0.3823394	0.0231027	0.3381893	0.4285212
25:p B:Strata 2 Session	0.3632224	0.0226552	0.3200834	0.4086808
26:p B:Strata 2 Session	0.3736499	0.0229021	0.3299544	0.4195089
27:p B:Strata 2 Session	0.3639963	0.0231925	0.3198595	0.4105472
28:p B:Strata 2 Session	0.3923596	0.0240168	0.3464123	0.4402958
29:p B:Strata 2 Session	0.3450874	0.0226209	0.3021916	0.3906634

and compare it with the derived estimates of \hat{N}^s under the usual parameterization:

MSCRD Example analysis classic param				
Estimates of Derived Parameters				
Population Estimates of {global corr}				
Grp.	Sess.	N-hat	Standard Error	95% Confidence Interval
				Lower Upper
1	1	509.53995	24.785248	468.08049 566.19996
1	2	606.41940	24.704667	563.99887 661.51256
1	3	642.06561	24.712968	599.96654 697.59483
1	1	486.29651	20.913781	451.40402 534.22362
1	2	569.23072	19.664402	535.67399 613.34348
1	3	634.77481	22.950855	595.70761 686.38228

For primary period 1, you can see that

$$\hat{\omega}_t^A = \frac{\hat{N}_t^A}{\hat{N}_t^A + \hat{N}_t^B} = \frac{509.54}{509.54 + 486.29} = 0.512,$$

and therefore the estimates for ω in the first output make sense, and under that parameterization can be modeled as a function of covariates.

One thing to note is that because the estimation of ω is based on the information from within each primary period, it can only apply to observable states. That is, in this example if there had been a third state, C , which was unobservable, one would just fix $\omega_t^C = 0$.

16.10.2. Analyzing data for just one primary period

You might be interested in focusing on analysis of just one primary period. One reason might be to estimate the abundance or residence time parameters discussed above. Another use for this approach is in model selection. Robust design models add a layer of complexity to model selection, because possible variation in parameters goes on both at the primary period and secondary period levels. One approach to simplifying this process is to at least partially partition model selection with respect to p_{ent} , φ , and p from model selection with respect to S and ψ . Regardless, if you are interested in analyzing data for a given primary period you have two choices. If you are willing to assume that φ is *not* a function of an animal's 'age' within the primary period, then you are dealing with a Schwarz-Arnason Jolly-Seber model, and you can use the POPAN option in MARK (Chapter 12).

Otherwise you need to 'trick' the MSORD model. To do this, pretend that you have a two-primary period study. The data you are interested in analyzing will constitute the first primary period, and you will create a dummy second primary period consisting of at least two capture occasions.

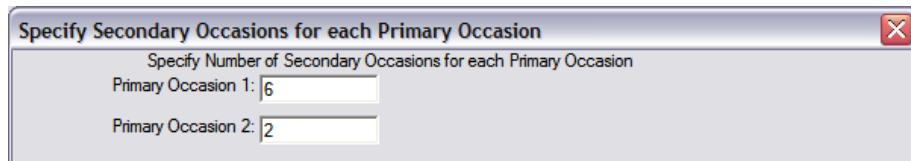
For example, if you want to analyze the following sea turtle capture histories for one primary period

```
111100
010010
111000
111100
011101
```

Then concatenate two more columns consisting of all 1's:

```
11110011 1;
01001011 1;
11100011 1;
11110011 1;
01110111 1;
```

Create an 'MSORD' model in MARK as discussed before. In this case you specify 8 total encounters, and using the 'Easy Robust Times' option, specify 2 primary periods, with 6 and 2 secondary samples, respectively.



You specify two states, as discussed above. Set up the PIM's as described above, with the following exception. In order for each of the animals captured in primary period 1 to have a history of '11' in primary period 2, each must have survived, returned to the study area in primary period 2, arrived in

time for the first sample, stayed around for each of the two sampling occasions, and been captured each time. Therefore, for the observable state you would need to fix the $S_1, \psi_1^{\text{obs} \rightarrow \text{obs}}, \varphi_{20}^{\text{obs}}, p_{21}^{\text{obs}}, p_{22}^{\text{obs}}$ and to 1.0 and p_{22}^{obs} to 0 (recall that is computed by subtraction). Maintain these constraints for each of the models you consider.

16.11. The robust design & unequal time intervals

As noted in Chapter 10, any data type with state transitions potentially suffers from the same problem when the intervals between occasions are unequal (how **MARK** handles unequal intervals in general was introduced earlier in Chapter 4).

As introduced in Chapter 10, consider the case where an encounter occasion is missing in the multi-state data type. Consider the following valid **MARK** 5-occasion multi-state encounter history ‘A.A00’, where the missing occasion is shown as a ‘dot’ and there are 2 states, **A** and **B**, and occasions are all 1 time unit apart. To explain this ‘dot’, several possibilities exist, namely:

$$S_1^A \psi_1^{AA} (1 - p_2^A) S_2^A \psi_1^{AA} p_3^A \dots \quad \text{and} \quad S_1^A \psi_1^{AB} (1 - p_2^B) S_2^B \psi_2^{BA} p_3^A \dots$$

However, suppose that you coded the data with the dot left out, and set the time intervals to 2, 1, and 1. That is, only 4 occasions are considered instead of 5. So the encounter history is now ‘AA00’. Unfortunately, this approach is going to give *very* different results from the proper parametrization above. **MARK** does not generate the probabilities for the transition to state **B** with this parametrization. The probability of surviving from occasion 1 to occasion 2 would now be $(S_1^S)^2$, with no consideration that the animal could have moved to state **B** during the missing occasion. So, even the survival estimates S will be incorrect. The ψ parameters for the first interval are not comparable to the ψ parameters for the second and third intervals because they represent different time scales.

Internally, within **MARK**, the time interval correction on S remains, but all time interval corrections from ψ have been removed. The motivating logic is that when time intervals are ‘ragged’, e.g., 1.1, 0.9, 1.05, 0.95, it may still make sense to apply a correction to S . However, this correction is inappropriate for ψ , and may even be questionable for S .

Given the deep connections between ‘multi-state’ models and the ‘robust design’ models introduced in this chapter, it is perhaps not surprising that the same general issue applies here. Consider the robust design with 3 primary occasions, each with 2 secondary occasions. Assume that the data were not collected for the 2nd primary sample, giving an encounter history of ‘11..11’.

The missing primary encounter history again can be explained by 2 possibilities:

$$\dots S_1 \gamma_2'' S_2 (1 - \gamma_2') \dots \quad \text{and} \quad S_1 (1 - \gamma_2'') (1 - p_2^*) S_2 (1 - \gamma_3'') \dots$$

For the robust design data type, coding the encounter history as only 2 primary occasions, ‘1111’, with time interval of 2 will give the correct parametrization for S (i.e., S^2), but as above, the γ' and γ'' parameters cannot be corrected with this simple trick because the possibility of leaving the encounter area is not considered. So, for robust design data types (including the multi-state robust designs), survival rates are corrected with the time interval. But what about the transition probabilities? Following from section 10.6 in Chapter 10, it might be reasonable to assume that the ‘transition’ parameters are not adjusted for interval length, since they aren’t for a ‘typical’ multi-state model.

However, the transition parameters of the robust design (as well as the Barker and Barker robust design models) are now corrected for time interval length. Although the robust design model can be equivalent to the multi-state model, correction for unequal time intervals is handled differently.

Robust design models assume survival is the same regardless of whether the animal is available or not available for capture. Thus, an animal can move between these 2 states and survival is not affected. This assumption of identical survival is required to get parameter identifiability in the model.

In contrast, the multi-state models only allow movement at the end of the time interval because survival is usually different between the 2 states, and thus movement during the interval would require a model with a weighted product of survival rates depending on timing of when movement occurred. Therefore the ψ parameters of the multi-state models do not need correction for the length of the time interval because they are in theory instantaneous, whereas the γ' and γ'' parameters of the robust design models should be corrected for the length of the time interval because you would expect more movement occurring during longer intervals.

In the robust design models, survival (S) is corrected for time interval length (L) as S^L . The γ' parameter is corrected as $(\gamma')^L$ and the γ'' parameter is corrected as $1 - (1 - \gamma'')^L$. The reason for this difference is that γ' is the probability of remaining unavailable for capture, and so is the probability of remaining off site (Fig. 16.5). This scenario is like survival, where S is corrected as S^L . In contrast, γ'' is the probability of becoming unavailable (Fig. 16.5), so that $(1 - \gamma'')$ is the probability of remaining on site. Thus $(1 - \gamma'')$ is the quantity needing correcting.

[In the Barker model, survival (S) is corrected for time interval length (L) as S^L . The R and R' parameters are corrected as $1 - (1 - R)^L$ and $1 - (1 - R')^L$, which puts each of these parameters on a 1-unit time scale. The F parameter is corrected as F^L and the F' parameter is corrected as $1 - (1 - F')^L$. The r parameter does not need a time correction because it only applies when the animal dies, which can only happen once no matter how long the time interval. Likewise, the p parameter does not need a time interval correction.]

In the Barker robust design model, survival (S) is corrected for time interval length (L) as S^L . The R and R' parameters are corrected as $1 - (1 - R)^L$ and $1 - (1 - R')^L$, which puts each of these parameters on a 1-unit time scale. The F parameter is also corrected as F^L . The a'' parameter is corrected as $(a'')^L$ and the a' parameter is corrected as $1 - (1 - a')^L$. The r parameter does not need a time correction because it only applies when the animal dies, which can only happen once no matter how long the time interval.

16.12. References

- Cormack, R. M. (1964) Estimates of survival from the sighting of marked animals. *Biometrika*, **51**, 429-438.
- Fujiwara, M., and Caswell, H. (2002) Temporary emigration in mark-recapture analysis. *Ecology*, **83**, 3266-3275.
- Huggins, R. M. (1989) On the statistical analysis of capture-recapture experiments. *Biometrika*, **76**, 133-140.
- Huggins, R. M. (1991) Some practical aspects of a conditional likelihood approach to capture experiments. *Biometrics*, **47**, 725-732.
- Jolly, G. M. (1965) Explicit estimates from capture-recapture data with both death and immigration stochastic model. *Biometrika*, **52**, 225-247.
- Kendall, W. L., and Pollock, K. H. (1992) The Robust Design in capture-recapture studies: a review and evaluation by Monte Carlo simulation. Pages 31-43 in *Wildlife 2001: Populations*, D. R. McCullough and R. H. Barrett (eds), Elsevier, London, UK.
- Kendall, W. L., and Nichols, J. D. (1995) On the use of secondary capture-recapture samples to estimate temporary emigration and breeding proportions. *Journal of Applied Statistics*, **22**, 751-762.

- Kendall, W. L., Pollock, K. H., and Brownie, C. (1995) A likelihood-based approach to capture-recapture estimation of demographic parameters under the robust design. *Biometrics*, **51**, 293-308.
- Kendall, W. L., Nichols, J. D., and Hines, J. E. (1997) Estimating temporary emigration using capture-recapture data with Pollock's robust design. *Ecology*, **78**, 563-578.
- Kendall, W. L., and Bjorkland, R. (2001) Using open robust design models to estimate temporary emigration from capture-recapture data. *Biometrics*, **57**, 1113-1122.
- Kendall, W. L., and Nichols, J. D. (2002) Estimating state-transition probabilities for unobservable states using capture-recapture/resighting data. *Ecology*, **83**, 3276-3284.
- Kendall, W. L., Stapleton, S., White, G. C., Richardson, J. I., Pearson, K. N., and Mason, P. (2019) A multistate open robust design: population dynamics, reproductive effort, and phenology of sea turtles from tagging data. *Ecological Monographs*, **89**, 1-17. e01329. 10.1002/ecm.1329
- Nichols, J. D., and Coffman, C. J. (1999) Demographic parameter estimation for experimental landscape studies on small mammal populations. In *Landscape Ecology of Small Mammals*. G. W. Barrett & J. D. Peles, Eds. Springer-Verlag, NY. pp. 287-309.
- Schaub, M., Gimenez, O., Schmidt, B. R., and Pradel, R. (2004) Estimating survival and temporary emigration in the multistate capture-recapture framework. *Ecology*, **85**, 2107-2113.
- Schwarz, C. J., and Stobo, W. T. (1997) Estimating temporary migration using the robust design. *Biometrics*, **53**, 178-194.
- Schwarz, C. J., and Arnason, A. N. (1996) A general methodology for the analysis of capture-recapture experiments in open populations. *Biometrics*, **52**, 860-873.
- Seber, G. A. F. (1965) A note on the multiple recapture census. *Biometrika*, **52**, 249-259.

Addendum – the RD + open population abundance estimation

Suppose you have a data set consisting of annual (primary) encounters with a sample of marked individuals. Assume there is only one sampling occasion per year (i.e., only one secondary sample per primary occasion). While it is clear that some of the parameters which can be estimated using the robust design are not accessible (e.g., γ' , γ''), you might be wondering about estimation of abundance. You might conclude that without multiple closed secondary samples within each primary sample, that estimation of annual abundance is not possible.

That is not entirely correct. You can in fact estimate abundance from a time series of single samples, if you're willing to make a few assumptions. Recall from Chapter 15 that for a single sampling event, the canonical estimator for abundance is

$$\hat{N} = \frac{n}{p} = \frac{M_{t+1}}{p},$$

where the numerator ($n \equiv M_{t+1}$) is the number of unique individuals encountered, and the denominator (p) is the probability that any individual will be encountered. So, if you have the number of uniquely encountered individuals at each primary sampling occasion, and an estimate of the encounter probability, then deriving an estimate of abundance is relatively straightforward.

Consider for example, the male Dipper data set (`ed_males.inp`), which consists of 7 live encounter occasions. Let's quickly fit the Cormack-Jolly-Seber (CJS) open population model $\{\varphi_t, p_t\}$ to these data (this model is not a particularly good fit to these data, but it serves for this demonstration).

Here are the estimates of the encounter probabilities for each primary sampling occasion:

7:p	0.7173913
8:p	1.0000000
9:p	0.9093023
10:p	0.9274194
11:p	0.9358289
12:p	0.7637580

Recall that the final estimate of \hat{p}_7 is confounded with $\hat{\varphi}_6$, so we will ignore this final estimate – meaning, we won't be able to derive an estimate of abundance for the final occasion. So, estimates $\hat{p}_2 \rightarrow \hat{p}_6$ form the denominators for our abundance estimation expression.

What about the numerators – the number of uniquely encountered individuals (M_{t+1})? For some data types in **MARK** where abundance is estimated as a derived parameter (including the robust design), **MARK** provides these for you in the ‘full output’ listing. But, this is not the case for the CJS model.

However, **MARK** can in fact provide you with M_{t+1} , if you know where to look. Simply access ‘**Output** | **Input Data Summary**’ which will generate the reduced m -array (introduced in Chapter 5). The shaded column (below) tabulates the M_{t+1} values we need ($R(i) \equiv M_{t+1}$).

Occ.	R(i)	j= 2	3	4	5	6	7	Total
1	12	6	1	0	0	0	0	7
2	26		11	0	0	0	0	11
3	37			17	1	0	0	18
4	39				22	0	1	23
5	45					25	0	25
6	48						28	28
7	46							0

But, notice that we have 6 total estimates of encounter probability ($\hat{p}_2 \rightarrow \hat{p}_7$, although because of confounding, we'll ignore \hat{p}_7), while we have 7 values for M_{t+1} . The difference between the two reflects the fact that there is no estimate for \hat{p}_1 , correspond to $M_{t+1} = 12$ for the first occasion. In fact, this illustrates just one of the advantages of using the robust design sampling protocol – multiple secondary samples let you estimate the encounter probability (p^*) for all primary sampling periods, including the first one. For the CJS analysis, where we can't estimate \hat{p}_1 , we'll have to accept that we can't estimate abundance for primary occasion 1, and if we can't estimate \hat{p}_7 , then we can't estimate \hat{N}_7 .

We can now use a short **R** script to manually derive our estimates of abundance:

```
# enter the M(t+1) values - dropping the first and last one
Mt1 <- c(26,37,39,45,48);

# enter the estimated detection probabilities
p_hat <- c(0.7173913,1.0,0.9093023,0.9274194,0.9358289);

# generate abundance estimates
N_est <- Mt1/p_hat;
print(N_est)

[1] 36.24242 37.00000 42.89003 48.52174 51.29143
```

So, at the second primary sampling occasion, we estimate 36.24 male dippers, at the third primary sampling occasion, we estimate 37.00 male dippers, and so on.

While this isn't particularly difficult, it is cumbersome to have to enter different values manually. And (more importantly), this approach makes it difficult to derive estimates of uncertainty for our derived estimates of abundance. It would be far more convenient if **MARK** could handle all this for us.

In fact, this is quite possible in **MARK** – you simply need to (i) re-format the .INP file such that there are 2 secondary samples per primary sampling occasion, and then (ii) analyze the re-formatted data in **MARK** as a robust design problem.

You might be puzzled by step (i) – how can you add a secondary sample to each primary period when in fact there is no secondary sample in the first place? Simple – we'll add a ‘dot’. Recall from earlier chapters that in **MARK**, we can use a ‘dot’ in the .INP file to represent a missing sampling occasion. Here, we're going to use the ‘dot’ to represent a ‘missing’ secondary occasion. Of course, it is not really ‘missing’, but **MARK** doesn't differentiate between ‘missing’ and ‘not existing’. **MARK** simply recognizes that the ‘dot’ in the .INP file means no data for that occasion.

How does this work? Basically, we're going to take an encounter history consisting of a single ‘1’ or ‘0’ for each primary sample, and add a ‘dot’ after each character in the encounter history. For example, we reformat the encounter history ‘11010110’ as

‘11010110’ → ‘1.1.0.1.0.1.1.0.’

Here is a sample of re-coded encounter histories from the male Dipper data set:

<i>event history</i>		<i>RD coded history</i>
1111110	→	1.1.1.1.1.1.0.
1111000	→	1.1.1.1.0.0.0
1010000	→	1.0.1.0.0.0.0.
0010110	→	0.0.1.1.0.0.0.

The idea behind the reformatting is straightforward – but doing it by hand is time-consuming, and likely prone to error. With a little bit of programming skill, its not too difficult to get a computer to do the work. Here is a short **R** script that makes use of ‘regular expressions’.*

This script (which will re-format `ed_males.inp` → `ed_males_RD.inp`) works best if you first strip out any comments you might have in the `.INP` file:

```
# read in EH file -- make use of colClasses to read in EH as string
input.data <- read.table("C:\\\\Users\\\\MARK\\\\Desktop\\\\ed_males.inp",header = FALSE,
  col.names = c("EH","freq"),colClasses = c("character", "character"));

# now do the RegEx - append a dot to element of EH string
input.data$EH <- gsub("(.)", "\\\\1.", input.data$EH)

# write out the file
write.table(input.data,"C:\\\\Users\\\\MARK\\\\Desktop\\\\ed_males_RD.inp",col.names=FALSE,
  row.names = FALSE, quote = FALSE);
```

Once you have the newly formatted `ed_males_RD.inp` file, you can proceed with the analysis as if it were a robust design problem. We simply need to remember that the encounter history is now 14 occasions long, and there are 7 primary samples, each consisting of 2 secondary samples.

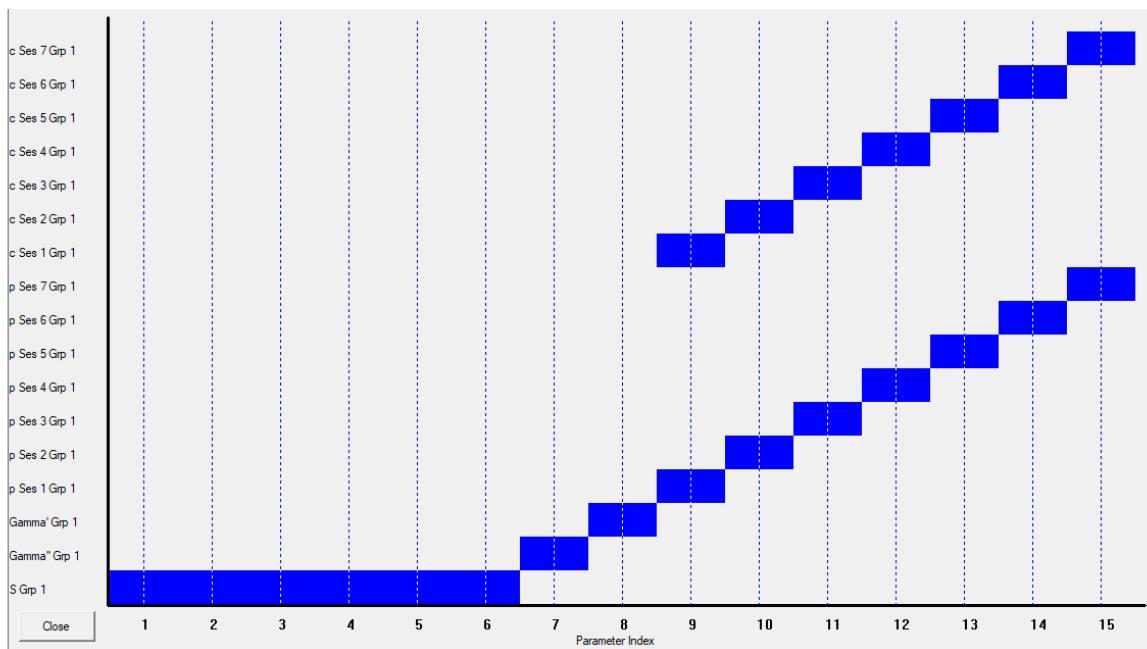
What about the other robust design parameters? In particular, what do we do with the γ' and γ'' parameters? Here, we'll simply assume no temporary emigration, and fix $\gamma' = 1$ and $\gamma'' = 0$. In principle, this would most closely match the assumptions underlying the CJS analysis, for model $\{\varphi_t p_t\}$.

For the encounter probabilities, p and c , we'll simply use model M_0 within each primary sample. Why? Because in fact we only have a single ‘true’ encounter occasion each year. The second occasion – the ‘dot’ – doesn’t really exist (and, in fact, with only 2 secondary occasions, real or otherwise, you cannot separately estimate p and c anyway). But, we will allow encounters to vary over time, among primary samples. We will use full time-dependence for survival, S (to match our CJS model). The PIM chart for our model – $\{S_t, \gamma' = 1, \gamma'' = 0, p = c\}$ – is shown at the top of the next page.

After fixing the γ parameters (as outlined earlier), here are the derived estimates of abundance (we’ve reformatted the output slightly to emphasize that we’re interested in estimates for occasions 2 → 6 only – the SE of 0 for estimates 1 and 7 are diagnostic of limitations discussed earlier):

Estimates of Derived Parameters					
Population Estimates of $\{S(t)(g=0,g'=1)\}$					
Grp.	Sess.	N-hat	Standard Error	95% Confidence Interval	
				Lower	Upper
1	1	26.661732	0.0000000	26.661732	26.661732
1	2	36.242427	11.923565	27.669417	88.840693
1	3	37.000000	0.1159867E-005	37.000000	37.000000
1	4	42.890027	4.5354597	39.632854	62.911217
1	5	48.521739	4.1182200	45.570878	66.725574
1	6	51.291430	3.8659007	48.530684	68.414246
1	7	60.228138	0.0000000	60.228138	60.228138

* For non-programmers, a ‘Regular Expression’, or ‘RegEx’, is a sequence of characters that forms a search pattern. Usually such patterns are used by string-searching algorithms for “find” or “find and replace” operations on strings, or for input validation for interactive forms (amongst many other uses). In the R script, the RegEx is coded in the second code block – ?gsub for details.



Compare these estimates with those we derived manually earlier:

```
[1] 36.24242 37.00000 42.89003 48.52174 51.29143
```

The results are identical (to within rounding error)* But, estimates using the robust design approach to the problem have the advantage of providing estimates of uncertainty for the estimates of abundance.

Now, back at the start of this addendum, we noted that the preceding works, conditional on your comfort with a few assumptions. Perhaps not ‘assumptions’ so much as recognizing the limits of your inference. If you’ve previously worked through Chapter 15, you will recall that modeling individual heterogeneity is the largest challenge in coming up with robust estimates of abundance. Even something as simple as $p \neq c$ can be an important consideration. In using the robust design approach described here, where we only have 1 true secondary sample per primary period, we have absolutely no way of accommodating heterogeneity. Period. So, to use the estimates, you’d need to assume that heterogeneity isn’t important – which for most taxa is probably not remotely the case. So, yes, you can estimate abundance from an open CJS model as demonstrated, but you need to be careful interpreting the robustness of those estimates. *Caveat emptor.*

* In addition, the estimates of survival from this RD analysis are identical to those from the original CJS analysis.