Public Health Monograph Series
No.25
ISSN 1178-7139

MODELLING OPTIONS FOR ABC-CBA

Burden of Disease Epidemiology, Equity and Cost-Effectiveness Programme (BODE^3)

Technical Report: Number 17

Giorgi Kvizhinadze
Tak Ikeda
Tony Blakely

May 2012

A technical report published by the Department of Public Health,
University of Otago, Wellington

ISBN 978-0-473-21124-0

ABC-CBA Team*
*** Contact Professor Tony Blakely (Principal Investigator of the ABC-CBA component of the BODE^3 Programme, University of Otago, Wellington, New Zealand) Email:** tony.blakely@otago.ac.nz

Acknowledgements

We thank other BODE^3 team colleagues for comments on early versions of this work, in particular Ken Richardson. We acknowledge Sophie Whyte for communications and prototype Metropolis-Hastings spreadsheets. The BODE^3 programme receives funding support from the Health Research Council of New Zealand (10/248).

Competing Interests

The authors have no competing interests.

Table of Contents

List of Tables

List of Figures

Report Objectives

The objectives of this report are to outline the modelling options for use in ABC-CBA, maintaining alignment with the BODE^3 imperatives to be able to model population heterogeneity (largely for equity analyses) and to retain fidelity with New Zealand data and burden of disease study foundations. This report gives 'in principle' options for modelling; all models will differ in their structure to some extent due to the nature of the intervention and/or research questions. The key issues addressed include:

- Markov models, including macro and micro-simulation variants
- Model calibration, in particular approaches to estimating transition probabilities (or time to event for DES) when full data is not available.

1 Introduction

The objectives of this report are to outline the modelling options for use in ABC-CBA, maintaining alignment with the BODE^3 imperatives to be able to model population heterogeneity (largely for equity analyses) and to retain fidelity with New Zealand data and burden of disease study foundations. This report gives 'in principle' options for modelling; all models will differ in their structure to some extent due to the nature of the intervention and/or research questions. The key issues addressed include:

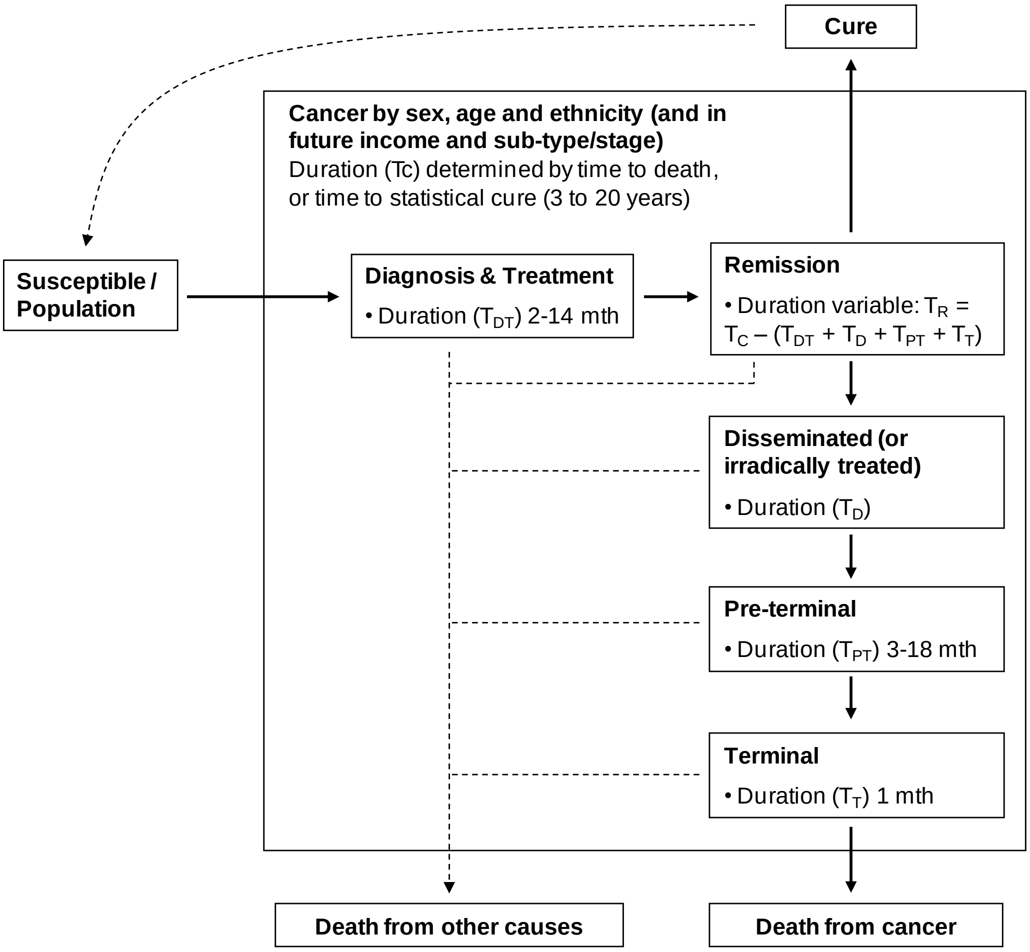
- Markov models, including macro and micro-simulation variants
- Model calibration, in particular approaches to estimating transition probabilities (or time to event for DES) when full data is not available.

Figure 1 shows a stylised cancer model, building on that used in burden of disease studies (BDS) (Begg, Vos et al. 2007; Blakely, Costila et al. 2010). Each cancer is assumed to have various states as depicted, with varying disability weights (DW; and hence quality of life) and duration for each state. In a New Zealand BDS, this model was used to estimate "years lived in disability" (YLDs) for incident cancers, once time of death was determined by a much simpler Markov model with just three states ([alive with cancer], [death from cancer], [death from other causes]) (Blakely, Costila et al. 2010). In principle it is possible to use the model depicted below in Figure 1 as a Markov model to estimate health adjusted life years (HALYs).[1] The problem is estimating the transition probabilities between each state, in that data on the time dependent transition probabilities is not observed. Rather, it is simply assumed retrospectively that if someone died of the cancer, they must have had some time prior to death in less than ideal states of terminal and pre-terminal. There are also a range of other models that could be used for estimating HALYs, and the consequences of various interventions; this report details some of these options. But our foremost objective is the specification of simple Markov models that are congruent with Figure 1 (Section 2).

[1] Whilst our methods originate from burden of disease methods, and one might therefore use the term 'DALYs averted', this can be confusing in that the measure is not a 'classic' DALY from a burden of disease study. Indeed, the only difference conceptually between what we estimate and a QALY from other cost effectiveness analyses is the use of disability weights instead of utilities.

Regardless of what specific disease model is used to estimate HALYs, an additional aim is to use sex-, age-, ethnicity-, deprivation- and stage/sub-type-specific inputs for incidence and survival from comprehensive baseline data assembled for BODE^3 (Costilla, Atkinson et al. 2011; Kvizhinadze and Blakely 2011; Blakely, Costilla et al. 2012). At a model development stage, we focus on monthly cycles – although we anticipate for many interventions in the future annual cycles may be sufficient.

Figure 1: General ABC-CBA cancer disease model, following burden of disease precedents


Cure
Cancer by sex, age and ethnicity (and in future income and sub-type/stage)
Duration (Tc) determined by time to death, or time to statistical cure (3 to 20 years)
Susceptible / Population
Diagnosis & Treatment
• Duration (T_{DT}) 2-14 mth
Remission
• Duration variable: $T_R = T_C - (T_{DT} + T_D + T_{PT} + T_T)$
Disseminated (or irradically treated)
• Duration (T_D)
Pre-terminal
• Duration (T_{PT}) 3-18 mth
Terminal
• Duration (T_T) 1 mth
Death from other causes
Death from cancer

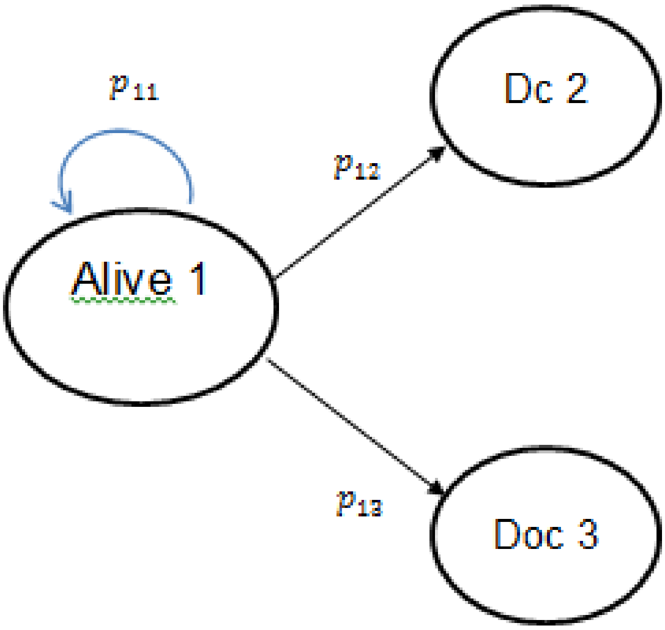
T_C= total cancer duration; T_{DT} = time in diagnosis and treatment state; T_R= time in remission state; T_D = time in disseminated state; T_{PT}= time in pre-terminal state; T_T= time in terminal state.

The structure of this Report is as follows. Section 2 outlines the simple Markov macrosimulation model that – with mathematical extensions – can capture the disease model shown in Figure 1, and satisfy the heterogeneity requirements of BODE^3. It is outlined as operationalized in TreeAge. A generic colorectal cancer intervention is used to demonstrate the approach. Section 3 then briefly details more complex Markov options, including microsimulation using Markov models. Section 4 briefly overviews DES options, a modelling option we anticipate being commonly used in BODE[3]. Finally, Section 5 outlines options to calibrate models with multiple states that are not fully observed, yet need transition probabilities specified that satisfy what is known overall about the disease. For example, we may have good data or estimates on overall cancer incidence and survival (including by socio-demographics), but poor data on the disaggregated transition probabilities between disease states (e.g. remission to pre-terminal; small adenoma to large adenoma).

2 Simple Markov model

We focus on a Markov model with 3 states, “Alive”, “Death from Cancer” (Dc), “Death from other causes” (Doc), the latter two being absorbing states. Equity will be explored initially at least by heterogeneity of inputs and findings across sub-populations, e.g. by sex, age and ethnicity. More immediately, many evaluations will require modelling stages of cancer (i.e. usually SEER stage from the cancer registry) separately, as each stage has very different survival and stage distribution also varies by socio-demographics. We will model their “experience” from the moment of diagnosis till death or the age of 110. Therefore, the maximum number of monthly cycles in the Markov model will be equal to 12× (110 – age at diagnosis).

Figure 2: Simple ABC-CBA Markov cancer disease model



$p_{11}(i), p_{12}(i), p_{13}(i)$ are calculated as follows. Suppose $m_1\ i$ and $m_2\ i$ are death rates by cycle (assumed 1 month) from cancer and from other causes respectively. Then $m_1\ i\ +\ m_2\ i\ = m\ i$ is the total death rate. Consequently total death probability is equal to:

$$p\ \ i\ \ = 1 - \exp(-m\ \ i\ \ )$$

$$p_{12}\ \ i\ \ = \frac{m_1\ \ i}{m\ \ i}\ p\ \ i$$

$$p_{13}\ \ i\ \ = \frac{m_2\ \ i}{m\ \ i}\ p\ \ i$$

$$p_{11}\ \ i\ \ = 1 - p_{12}\ \ i\ \ - p_{13}\ \ i$$

A limitation of such a simplified Markov model is that it is difficult to attach heterogeneous and time varying effectiveness to each state for each cycle (e.g. disability weights, cost to health service of being in that state). Time varying effectiveness can be captured with large look-up tables that use the cycle number to direct which value is taken. A more challenging issue, however, is estimating the HALYs from a simple Markov model given the desired disease model in Figure 1**.**

2.1 Simple Markov model implemented in TreeAge

In the "Cost-effectiveness" calculation mode, TreeAge allows one to assign cost and effectiveness to given states. There are three types of costs and effectiveness that can be assigned: initial, incremental and final cost and effectiveness. A state's initial cost and effectiveness (if any) is assigned only in the first cycle and only to the individuals that spend the first cycle in that state. The incremental cost and effectiveness is assigned in the subsequent cycles during the process. The final cost and effectiveness (if any) is assigned after the process is over to individuals ending up in that state. For example, let the initial cost reward be $2000, the incremental cost reward be $1000 and the final reward be $500 for the state 'alive'. Then all people starting the model in the alive state will receive a $2000 reward at the end of the first cycle, and everyone in the alive state in the second cycle and beyond will receive $1000 per cycle. Everyone in the alive state in the final cycle of the model (however long that be e.g. on turning 110 years of age in our models) will receive a $500 reward if still alive.

The cohort (individual in case of microsimulation) diagnosed with cancer enters in the state "Alive" at "0 stage"[2]. Everyone is alive in the cycle 1. At the end of cycle 1 (start of cycle 2) some portion die of cancer (i.e. jump into "Dc" state), some proportion dies from other causes (i.e. jump into "Doc" state), and the rest stay alive (stay in "Alive" state). The model also contains the state "Dead" which we use as "dummy" state to simplify calculations. In the Cost-Effectiveness mode each state is attributed certain costs and effectiveness of being in that state. Note that it is important that no one stays in "Dc" and "Doc" states longer than 1 cycle to ensure that they generate "effectiveness" only once. This is the reason why we introduced the dummy state "Dead".

2.2 Example: Colorectal cancer intervention

To demonstrate the modelling, and presentation of outputs, we specified up a macrosimulation model for colorectal cancer, for ages 50-64 years (50-54, 55-59, and 60-64), with further stratification by ethnicity, gender and deprivation level. This gives 36 strata in total (i.e. the strata over which population heterogeneity can be considered). Baseline data were taken from the expected population distribution at 2011 of new colorectal cancer cases (i.e. from projected cancer incidence rates multiplied into projected population distribution in 2011). Projected lifetables were used for background mortality rates (Kvizhinadze and Blakely 2011). Projected excess mortality rates were used for baseline cancer mortality (Blakely, Costilla et al. 2012).

Two simple interventions were simulated:

1) Intervention that reduces cancer mortality rate by 20% with direct costs of $1000 per month (cycle) in the first six months;
2) Intervention that reduces cancer mortality rate by 40% with direct costs of $5000 per month in first six months.

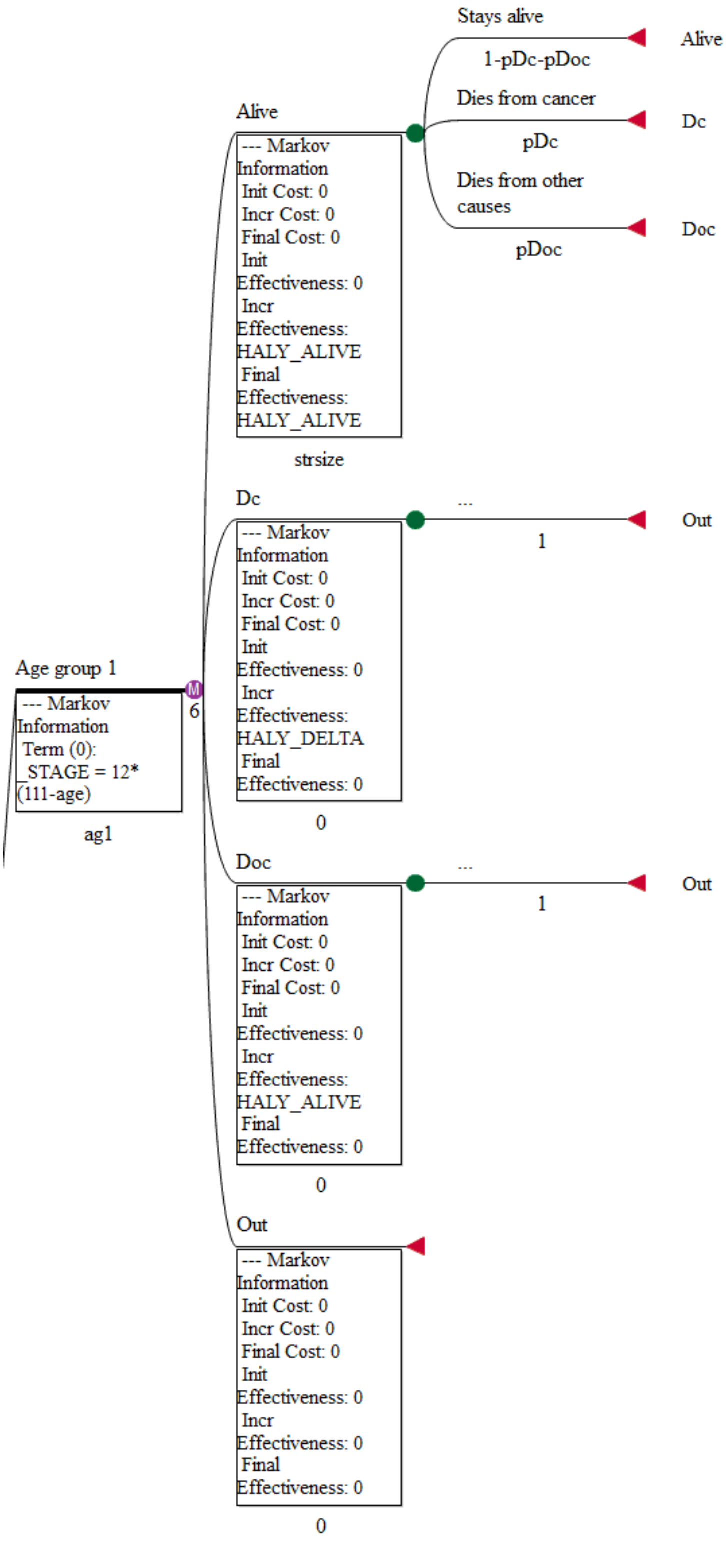
The Markov node consists of 4 states, "Alive" (person is alive), "Dc" (person died from cancer), "Doc" (person died from other causes) and "Out" (people are taken out of the

[2] TreeAge has built-in counter *_stage* which is equal to 0 before first cycle starts.

[2] Different cancers are relevant to different age groups.

model). The number of people in each strata depend on the incidence rate and the population count, and the number of cycles (months) in the Markov chain depends on the age at diagnosis. Everyone enters the “Alive” state and during each cycle they either stay in “Alive”, move into “Dc” or move into “Doc”. This movement is governed by the cancer excess mortality and background mortality.

Figure 3: Generic Markov model


Stays alive
1-pDc-pDoc
Alive
Dies from cancer
pDc
Dc
Dies from other causes
pDoc
Doc
Alive
--- Markov Information Init Cost: 0 Incr Cost: 0 Final Cost: 0 Init Effectiveness: 0 Incr Effectiveness: HALY_ALIVE Final Effectiveness: HALY_ALIVE
strsize
Dc
--- Markov Information Init Cost: 0 Incr Cost: 0 Final Cost: 0 Init Effectiveness: 0 Incr Effectiveness: HALY_DELTA Final Effectiveness: 0
0
...
1
Out
Age group 1
--- Markov Information Term (0): _STAGE = 12*(111-age)
ag1
M
6
Doc
--- Markov Information Init Cost: 0 Incr Cost: 0 Final Cost: 0 Init Effectiveness: 0 Incr Effectiveness: HALY_ALIVE Final Effectiveness: 0
0
...
1
Out
Out
--- Markov Information Init Cost: 0 Incr Cost: 0 Final Cost: 0 Init Effectiveness: 0 Incr Effectiveness: 0 Final Effectiveness: 0
0

Note that there are no (i.e. zero) rewards assigned in this skeletal model. See text for details of cost and effectiveness rewards.

The background mortality rates were picked up from the lookup table *bTable* containing 1332 rows and 6 columns. The first column contains the 'true' background mortality rates, the second column contains prevalent disability weights (pYLD).[3] The third column (bmr1) contains background mortality rates, assuming Māori and NZE have equal rates (i.e. that of NZE). The fourth column (pYLD1) contains pYLDs, assuming Māori and NZE have equal prevalent YLDs. The fifth column (pcount) contains the population counts for specific strata. The sixth column contains the cancer incidence rates for specific strata. The population counts are derived from as follows: NZE and Māori population estimates by gender and age for 2011 are obtained from statistics NZ website which is the disaggregated by deprivation tertiles assuming proportions by deprivation for 2011 is same as proportions by deprivation in 2006. Cancer incidence rates are obtained from (Costilla, Atkinson et al. 2011).

The number of rows is derived from the combination ethnicity, gender, deprivation and age: 1332 = 2(ethnicity) × 2(gender) × 3(deprivation) × 111(age). The table of background mortality rates, *bTable*, is arranged as shown in Table 1, with the first column listing ethnicity, the second column lists gender within each ethnicity, the third column lists three deprivation groupings within each gender group, final column lists age from 0 to 110 within each deprivation group.

Table 1: The structure of background mortality rates table (numbered columns are as per column numbering in lookup table)

Ind	**Ethn**	**Gen**	**Dep**	**Age**	**1.bmr**	**2.pYLD**	**3.bmr1**	**4.pYLD1**	**5.pcount**	**6.irate**
1	NZE	Male	Dep I	0	0.003537	0.036115	0.003537	0.036115	38503.43	0
2	NZE	Male	Dep I	1	0.000319	0.016217	0.000319	0.016217	38503.43	0
112	NZE	Male	Dep II	0	0.004996	0.036115	0.004996	0.036115	44971.91	0
334	NZE	Female	Dep I	0	0.002589	0.031375	0.002589	0.031375	36971.92	0
667	Māori	Male	Dep I	0	0.003676	0.054173	0.003537	0.036115	6072.701	0
1161	Māori	Female	Dep II	50	0.004138	0.128812	0.002033	0.085875	5821.465	36.0218
1332	Māori	Female	Dep III	110	1.110239	0.586897	1.019883	0.391265	139.8305	380.728

[3] pYLDs are prevalent YLDs. They are sourced from a burden of disease study, and are the average morbidity in the population, by sex, age (and possibly ethnicity and deprivation). That is, they are the average DW across all prevalent diseases for the given socio-demographic strata. The reason for including pYLDs in the modelling is that 1 minus pYLD is the total envelope of good health that can be enjoyed (on expectation) for any given strata, and as populations age this become considerably less than perfect (i.e. < 1).

In order to pick up the appropriate background mortality rates (bmr column) from the table *bTable*, we construct the function ind. First, let us define the following variables: *cycle*, *i_ethn*, *i_gen*, *i_dep*.

The text in bold italics equates to TreeAge inbuilt functions. e.g. ***_stage,*** the text in italics are the variables introduced by the modeller.

$$cycle = \boldsymbol{\_stage} + 1$$

Where ***_stage*** is TreeAge inbuilt stage counter starting from stage 0, hence *cycle* becomes the cycle counter starting from 1.

$$i\_ethn = \begin{matrix} 0, for\ \textbf{NZE} \\ 666, for\ \textbf{Maori} \end{matrix}$$

i_ethn is the increment of the index in the table *bTable* when ethnicity is Māori.

$$i\_gen = \begin{matrix} 0, for\ \textbf{MALE} \\ 333, for\ \textbf{Female} \end{matrix}$$

i_gen is the increment of the index in the table *bTable* when gender is female.

$$i\_dep = \begin{matrix} 0, for\ \textbf{Deprivation 1\_3} \\ 111, for\ \textbf{Deprivation 4\_7} \\ 222, for\ \textbf{Deprivation 8\_10} \end{matrix}$$

i_dep is the increment of the index in the table *bTable* corresponding to deprivation.

Then, in the Markov nodes we define the function *ind* which returns index for given ethnicity, gender, deprivation, age and cycle. We use TreeAge integrated Python programing language to define *ind*:

```
def ind(cycle,age,i_dep,i_gen,i_ethn):
return 1+age+int((cycle-1)/12)+i_dep+i_gen+i_ethn
```

(***int*** in TreeAge is an inbuilt function which returns the integer part of number).

For example, the index of Māori/female/deprivation 4-7/ age=47 in $cycle$ 44 is calculated as follows

$$ind = 1 + 47 + \boldsymbol{int}\ \frac{44 - 1}{12}\ + 111 + 333 + 666 = 1161$$

Note that in Table 1 the age value is 50 which is exactly how old 47 year olds will be after 44 cycles (months). Also after 44 cycles the calendar year will increase by 3.

As life expectancy increases into the future, and mortality rates fall, we therefore incorporate projected future mortality rate decline in the Doc transition probabilities. As this reduction is assumed to the same percentage by calendar year (1.75% per year for non-Māori, 2.25% for Māori, up to 2026; then 1.75% per year for both ethnic groups beyond 2026), it is equivalent to discounting the background mortality rates. Therefore, the background mortality rates for a given ethnicity, gender, deprivation, age and cycle is equal to:

$\boldsymbol{Discount}(bTable\ ind; 1\ ; redrate; \mathbf{int}((cycle - 1)/12))$.

In other words, the value from the table bTable row equals to $\boldsymbol{ind}$ and the column equals to 1 is reduced by $redrate$ over the time equal to $\mathbf{int}((cycle - 1)/12)$. The calendar year of diagnosis is incorporated in the following equation : The background mortality rate in the given calendar year of diagnosis for the given ethnicity, gender, deprivation, age and cycle is equal to:

$$MR = \boldsymbol{Discount}(\boldsymbol{Discount}(bTable[\boldsymbol{user}("ind"; cycle; age; i\_dep; i\_gen; i\_ethn); 1]; redrate; \mathbf{int}((cycle - 1)/12)); redrate; year - 2006)$$

$(\boldsymbol{user}("ind"; cycle; age; i\_dep; i\_gen; i\_ethn)$ is the way to call Python defined function`s, $"ind"$, value for arguments $cycle, age, i\_dep, i\_gen, i\_ethn)$.

Again MR is defined in the Markov nodes, $redrate$ is a yearly reduction in background mortality rate equal to 1.75% for NZE and 2.25% for Māori respectively. Note that from 2026 we assume the Māori rate reduction switches to that of non-Māori.

$$redrate = \begin{matrix} \boldsymbol{if}(year \leq 2026; 0.0225; 0.0175), for\ \mathbf{Maori} \\ 0.0175, for\ \mathbf{NZE} \end{matrix}$$

$\boldsymbol{Discount}(x; r; t)$ is a TreeAge function that returns the discounted value of x with a discount rate r over time t. Inner "$\boldsymbol{Discount}$" in the formula assures that after every 12 cycles, the background mortality rate is reduced by $redrate$, regardless of the calendar year of diagnosis. Outer "$\boldsymbol{Discount}$" takes into account the effect of calendar year of diagnosis.

So MR returns the background monthly mortality rate for a given calendar year, ethnicity, gender, deprivation, age and cycle.

Other important variables that use the function ind are $pcount$ and $irate$. The function $pcount$, population count for specific strata, is defined as follows:

$$pcount = bTable[User("ind"; 0; age; i\_dep; i\_gen; i\_ethn); 5]$$

The function $irate$, the cancer incidence rate for specific strata, is defined as follows:

$$irate = bTable[User("ind"; 0; age; i\_dep; i\_gen; i\_ethn); 6]$$

Finally we define the function $strsize$, which calculates the number of cancer cases in the given strata.

$$strsize = \frac{irate * pcount}{100000}$$

The functions $irate, pcount,$ and $strsize$ are defined in the Markov nodes, $strsize$ is assigned as a transition probability from the Markov node to the state "Alive".

The variable EMR is the excess mortality rate as a function of strata defining socio-economic indicators. Poisson and Negative Binomial regression models were used (Atkinson, Blakely et al. 2012; Blakely, Costilla et al. 2012) to predict cancer excess mortality rate for given calendar year, ethnicity, gender, deprivation, age and cycle (follow up time). Cubic splines were used to obtain predictions for continuous follow up time. The reference strata in the regression models is NZE/Male/Deprivation1-3/Age group 40-45/calendar year 2011. We used the regression equations within TreeAge rather than using large lookup tables. These calculations require introduction of regression model parameters from the excess rate modelling, defined in the Markov node. They are, main effects: r_Māori, r_female, r_dep4_7, r_dep8_10, r_diagnyear, r_ag45_54, r_ag55_64, r_ag65_74, r_ag75_110, interactions: r_int_65_74_firstyear, r_int_65_74_secondyear, r_int_75_110_firstyear, r_int_75_110_secondyear. We use 3 interior knots for cubic splines, (5 including exterior knots), r_k1 (lower exterior knot, always corresponds to one month after diagnosis), r_k2,r_k3,r_k4, r_k5 (upper exterior knot, always corresponds to statistical cure time). Three interior knots requires 5 coefficients (including the slope, constant) r_const, r_endb1, r_endb2, r_endb3, and r_endb4.

The cubic spline also require 4 basis functions *r_z1*, *r_z2*, *r_z3*, *r_z4*. They are defined in the root node **CRC** as follows:

$$r\_z1 = cycle/12$$

$$r\_z2 = \boldsymbol{max}(0; (cycle/12 - r\_k2)\text{^}3) - \boldsymbol{max}(0; (r\_k5 - r\_k2) * (cycle/12 - r\_k1)\text{^}3/(r\_k5 - r\_k1)) - \boldsymbol{max}(0; (1 - (r\_k5 - r\_k2)/(r\_k5 - r\_k1)) * (cycle/12 - r\_k5)\text{^}3)$$

$$r\_z3 = \boldsymbol{max}(0; (cycle/12 - r\_k3)\text{^}3) - \boldsymbol{max}(0; (r\_k5 - r\_k3) * (cycle/12 - r\_k1)\text{^}3/(r\_k5 - r\_k1)) - \boldsymbol{max}(0; (1 - (r\_k5 - r\_k3)/(r\_k5 - r\_k1)) * (cycle/12 - r\_k5)\text{^}3)$$

$$r\_z4 = \boldsymbol{max}(0; (cycle/12 - r\_k4)\text{^}3) - \boldsymbol{max}(0; (r\_k5 - r\_k4) * (cycle/12 - r\_k1)\text{^}3/(r\_k5 - r\_k1)) - \boldsymbol{max}(0; (1 - (r\_k5 - r\_k4)/(r\_k5 - r\_k1)) * (cycle/12 - r\_k5)\text{^}3)$$

We then introduced auxiliary parameters r_ethn, r_gen, r_dep, r_ag in the Markov node, defined as follows:

$$r\_ethn = \begin{matrix} r\_maori, for\ \textbf{Maori} \\ 0, for\ \textbf{NZE} \end{matrix}$$

$$r\_gen = \begin{matrix} r\_female, for\ \textbf{Female} \\ 0, for\ \textbf{Male} \end{matrix}$$

$$r\_dep = \begin{matrix} 0, for\ \textbf{Deprivation 1\_3} \\ r\_dep4\_7, for\ \textbf{Deprivation 4\_7} \\ r\_dep8\_10, for\ \textbf{Deprivation 8\_10} \end{matrix}$$

$$r\_ag = \begin{matrix} 0, if\ age < 45 \\ r\_ag45\_54, if\ age \in [45; 54] \\ r\_ag55\_64, if\ age \in [55; 64] \\ r\_ag65\_74, if\ age \in [65; 74] \\ r\_ag75\_110, if\ age > 74 \end{matrix}$$

$$r\_int\_followup = \begin{matrix} 0, if\ age < 45 \\ 0, if\ age \in [45; 54] \\ 0, if\ age \in [55; 64] \\ r\_int\_65\_74\_firstyear * \boldsymbol{if}(cycle \leq 12; 1; 0), if\ age \in\ \ 65; 74\ \ and\ cycle \leq 12 \\ r\_int\_65\_74\_secondyear * \boldsymbol{if}(12 \leq cycle \leq 24; 1; 0), if\ age > 74 if\ age \in\ \ 65; 74\ \ and\ 12 \leq cycle \leq 24 \\ r\_int\_75\_110\_firstyear * \boldsymbol{if}(12 \leq cycle \leq 24; 1; 0), if\ age > 74\ and\ cycle \leq 12 \\ r\_int\_75\_110\_secondyear * \boldsymbol{if}(12 \leq cycle \leq 24; 1; 0), if\ age > 74\ and\ and\ 12 \leq cycle \leq 24 \end{matrix}$$

$$r\_year = r\_diagnyear * (year - 2011)$$

In the Markov nodes we define the function which calculates the predicted excess mortality rate for given strata. Suppose T is a statistical cure time, that is the time after which, by our assumption, excess mortality rate becomes 0, then:

$EMR = \boldsymbol{if}(cycle <= T; 1; 0) * \mathbf{EXP}(r\_const + r\_endb1 * r\_z1 + r\_endb2 * r\_z2 + r\_endb3 * r\_z3 + r\_endb4 * r\_z4 + r\_ethn + r\_gen + r\_dep + r\_year + r\_int\_followup)$

The next step is to define transition probabilities from “Alive” state to “Dc” and “Doc” states in the Markov nodes, denote these probabilities with pDc and $pDoc$ respectively.

$$pDc = \boldsymbol{RateToProb}(es * EMR + MR; 1/12)\frac{es * EMR}{es * EMR + MR}$$

$$pDoc = \boldsymbol{RateToProb}(es * EMR + MR; 1/12)\frac{MR}{es * EMR + MR}$$

Where $\boldsymbol{RateToProb}(\boldsymbol{r}; \boldsymbol{t})$ is TreeAge function that returns probability over time t when the rate is equal to r and es is parameter which incorporates an effect size for the intervention. (We discuss these parameters later). The parameters, variables and functions described above are equivalent to parameters used to populate multistate life tables. In the following paragraph we introduce parameters, variables and functions that are used to calculate effectiveness (HALYs).

Table 2: Regression model parameters used in the colorectal cancer example

Parameter	Description	Value
r_const	Coefficient γ_0	-1.086
r_endb1	Coefficient γ_1	-1.199
r_endb2	Coefficient γ_2	-0.218
r_endb3	Coefficient γ_3	0.198
r_endb4	Coefficient γ_4	-0.074
r_k1	Position of first knot	1/12
r_k2	Position of second knot	1.6666
r_k3	Position of third knot	3.3333
r_k4	Position of fourth knot	5.4166
r_k5	Position of fifth knot	8
r_diagnyear	Year of diagnosis effect	-0.009
r_Māori	Ethnicity effect	0.435
r_female	Sex effect	-0.067
r_ag45_54	Age group 45-54 effect	0.183
r_ag55_64	Age group 55-64 effect	0.171
r_ag65_74	Age group 65-74 effect	0.124
r_ag75_110	Age group 75+ effect	-0.025
r_dep4_7	Deprivation 4-7 effect	0.059
r_dep8_10	Deprivation 8-10 effect	0.071
r_int_65_74_firstyear	Interaction of age group 65-74 with 1st year of diagnosis	0.071
r_int_65_74_secondyear	Interaction of age group 65-74 with 2nd year of diagnosis	0.215
r_int_75_110_firstdyear	Interaction of age group 75+ with 1st year of diagnosis	0.586
r_int_75_110_secondyear	Interaction of age group 75+ with 2nd year of diagnosis	0.359

2.2.1 Calculating HALYs

The first set of parameters we introduce in this section are disease model parameters or model structure parameters (Table 3). Times spent in health states: T_T, T_PT, T_DT and T (already introduced in the previous section) times in terminal, pre-terminal, diagnosis & treatment, statistical cure time respectively. Disability weights associated with these health states DW_T, DW_PT, DW_DT, DW_R, are disability weights in terminal, pre-terminal, diagnosis & treatment and remission states respectively. All these parameters are defined in the CRC node. In calculating HALYs we assume that everyone is awarded HALYs by cycle as if they are going to survive the cancer. That is we assume that they get utilities corresponding to the diagnosis & treatment state within T_DT months after diagnosis, then they get utilities corresponding remission state for all following months until statistical cure time T (or death from another cause). If they die from cancer, however, we modify the HALYs given. Namely we add HALYs assuming the last month was spent in the terminal state and T_PT months were spent in pre-terminal state before terminal state and subtract HALYs for previous T_T+T_PT months while assuming that they were going to survive. Those who die from background mortality we award survivors HALYs truncated at the moment of death, that is, we disregard possible higher background morbidity for those dying of another cause (assuming it is captured 'on average' in the pYLDs).

Table 3: Model input parameters for the colorectal cancer example

Parameter	**Description**	**Values, Range, Distribution**
DW_DT	Disability weight of DT	0.43
DW_R	Disability weight of RT	0.25
DW_PT	Disability weight of PT	0.83
DW_T	Disability weight of T	0.93
t_T	Time in T	1 month
t_PT	Time in PT	3 months
t_DT	Time in DT	9 months
T	Statistical cure time	8 years, 96 months
Rdisc	Remission DW discount rate (annual)	0.20%
redrateM	Background mortality yearly reduction rate for Māori	0.0225 (before 2026) 0.0175 (after (2026)
redrateNZE	Background mortality yearly reduction rate for Non-Māori	0.0175
discrateC	Discount rate for costs	0.03
discrateH	Discount rate for HALYs	0.03

Parameter	Description	Values, Range, Distribution
cos_within_first year	Monthly cost within first year of diagnosis	1000
cos_after_first year	Monthly cost after first year of diagnosis	200
cos_sur	Monthly cost (to health sector) after statistical cure time	100
cos_Dc_last year	Monthly cost within last year of life if died from cancer	500
cos_Doc_last year	Monthly cost within last year of life if died from other causes	400
cos_dir_1	Monthly direct cost of intervention 1, applied to only first 6 months after diagnosis	Gamma(mean=1000, SD=200)
cos_dir_2	Monthly direct cost of intervention 2, applied to only first 6 months after diagnosis	Gamma(mean=5000, SD=500)
es_1	Effect size of intervention 1	Normal(0.8, 0.05)
es_2	Effect size of intervention 2	Normal(0.6, 0.04)

2.2.1.1 Effectiveness for state “Alive”

Beside disease disability weights each individual experiences background disability weight mainly due to age, but it may also vary by gender, ethnicity, deprivation level or other socio-economic factors, this is called prevalent disability weight (pYLD). This is another strata dependent parameter of the model which we already included in *bTable* (Table 1) in the final column.

Now we define variable *DW*-disease related disability weight that individual experiences cycle by cycle (month by month), assuming that he/she is going to survive cancer and while in remission state his/her disability weight is reducing by roughly by 1.85% monthly. To define *DW*, we again use TreeAge integrated Python programing language to define *DW* as a function of *cycle*, *T_DT*, *DW_DT*, *DW_R and Rdisc*, this is done as follows:

```
DW=def DW(cycle,T_DT,T,DW_DT,DW_R,Rdisc):
    if cycle<=T_DT:
        return DW_DT
    elif T_DT<cycle<=T:
        return Discount(DW_R;Rdisc;cycle- T_DT-1)
    else:
        return 0
```

Where $Rdisc$ is monthly reduction of remission disability weight. The variable DW is defined in the Markov node**.** Then we calculate HALYs generated by individual as follows:

$$HALY\_ALIVE = \boldsymbol{Discount}((1 - bTable[\boldsymbol{User}("ind"; cycle - 1; age; i\_dep; i\_gen; i\_ethn); 2]) * (1 - \boldsymbol{User}("DW"; cycle - 1; T\_DT; T; DW\_DT; DW\_R; Rdisc)); discrate; (cycle - 1)/12)/12$$

Where $bTable[\boldsymbol{User}("ind"; cycle - 1; age; i\_dep; i\_gen; i\_ethn); 2]$ is the $pYLD.$ $HALY\_ALIVE$ is defined in each Markov node and will be assigned as an incremental and final effectiveness in the Markov state "Alive".

2.2.1.2 Effectiveness for state "Dc"

As described above, if an individual dies from cancer, we assume that the last month was spent in the terminal state and *T_PT* months were spent in pre-terminal state before terminal state. In these states we have already awarded *HALY_ALIVE* assuming that they were going to survive. This *HALY_ALIVE* needs to be adjusted to what it would have been due to terminal and pre-terminal states. The effectiveness awarded in the state "Dc" as incremental effectiveness are defined in the "Dc**"** node.

$$
\begin{aligned}
HALY\_DELTA = (&\boldsymbol{Discount}((1 - bTable[\boldsymbol{User}("ind"; cycle \\
&- 1; age; i\_dep; i\_gen; i\_ethn); 2]) * (1 - DW\_T); discrate; (cycle \\
&- 1)/12) + if(cycle \\
&> 2; 1; 0) * \boldsymbol{Discount}((1 - bTable[User("ind"; cycle \\
&- 2; age; i\_dep; i\_gen; i\_ethn); 2]) * (User("DW"; cycle \\
&- 2; T\_DT; T; DW\_DT; DW\_R; Rdisc) - DW\_PT); discrate; (cycle \\
&- 2)/12) + if(cycle \\
&> 3; 1; 0) * \boldsymbol{Discount}((1 - bTable[User("ind"; cycle \\
&- 3; age; i\_dep; i\_gen; i\_ethn); 2]) * (User("DW"; cycle \\
&- 3; T\_DT; T; DW\_DT; DW\_R; Rdisc) - DW\_PT); discrate; (cycle \\
&- 3)/12) + if(cycle \\
&> 4; 1; 0) * \boldsymbol{Discount}((1 - bTable[User("ind"; cycle \\
&- 4; age; i\_dep; i\_gen; i\_ethn); 2]) * (User("DW"; cycle \\
&- 4; T\_DT; T; DW\_DT; DW\_R; Rdisc) - DW\_PT); discrate; (cycle \\
&- 4)/12))/12
\end{aligned}
$$

2.2.1.3 Effectiveness for state "Doc"

HALYs for individuals who die from other causes is calculated based on the assumption that he/she was going to survive cancer. Therefore, we just award $HALY\_ALIVE$ in the last cycle of life. The rest of the HALYs generated by this individual were already awarded in the "Alive" state in the previous cycles. So $HALY\_ALIVE$ already defined in the **CRC** node will be assigned as incremental and final effectiveness in the Markov state "Doc".

2.2.2 Calculating costs

We now define parameters, variables and functions that are used to calculate the costs associated with baseline and intervention scenarios.

In ABC-CBA we distinguish two types of costs: direct cost of intervention and cost offsets.

Direct costs may be incurred at the outset, or by month alive, etc, and they may or may not vary in amount by cycle or other criteria. In this simple example, they are assumed to occur monthly in first six months only.

cos_dir – direct costs of intervention,

$$
cos\_dir = \begin{array}{ll}
\cos\_dir\_bsl, & in\ the\ node\ \textbf{Baseline} \\
\cos\_dir\_int1, & in\ the\ node\ \textbf{Intervention 1} \\
\cos\_dir\_int2, & in\ the\ node\ \textbf{Intervention 1}
\end{array}
$$

Cost offsets will be sourced from Health Tracker (Atkinson, Blakely et al. 2012). As described elsewhere, they will vary by socio-demographics, time since diagnosis, and whether in the last six or 12 months of life (final decision not made at time of this Report; differentiated by whether death from cancer of interest, or other causes). As described above, and detailed below, we use as simple specification in this colorectal cancer example:

$\cos\_within\_first\_year$ – cost offsets generated within first year of diagnosis, defined in the node **CRC**,

$\cos\_after\_first\_year$– cost offsets generated after first year of diagnosis, but before statistical cure time, defined in the node **CRC,** and

$\cos\_Dc\_last\_year$ – cost offsets generated by someone within a year of death from the cancer in question – namely CRC in this example. Note that we do not assign such costs retrospectively before the date of diagnosis for someone dying within 12 cycles of diagnosis. Rather, if someone died at the end of the 7th cycle, their cost-offsets are 'corrected' to be that for 7 cycles in the last year of life before cancer death, over-writing those already assigned by cycle assuming they were a survivor. Likewise, for people dying of cancer beyond the 12th cycle (but before the statistical cure time of course), their total cost offsets are retrospectively corrected to have 12 months in the last year of life rather than the assumed survival cost offsets. This is defined in the node **CRC.**

$\cos\_Doc\_last\_year$ – cost offsets generated by someone within a year of death from any cause other than the cancer in question. As with the above cancer death cost-offsets, a retrospective correction is applied at the cycle of death, to inflate the cost offsets accrued in the 11 previous cycles to now represent those for someone in the last year of life. Unlike cancer death cost-offsets, though, these death from other causes cost offsets apply after the statistical cure time. This cost is denoted with and defined in the node **CRC.**

$\cos\_sur$ – cost offsets generated after the statistical cure time. These three costs are assumed to be incurred by individuals that survive cancer and they will be applied to everyone alive (in the "Alive" state of Markov model). As stated above, we modify this cost if they die from other causes after the statistical cure time.

We now give more details on the exact mathematical calculations.

2.2.2.1 Costs for state “Alive”

The individual in the Markov state “Alive”, by our assumption, is going to survive cancer. That is, he/she incurs monthly cost $cos\_within\_first\_year$ during first 12 cycles of the Markov model, $cos\_after\_first\_year$ from cycle 13 up to statistical cure time and cos_sur after statistical cure time.

First we introduce some auxiliary parameters and functions, $discrateC$- discount rate for costs, defined in the **CRC** node. We specify a Python defined function cos that calculates the cost associated with each cycle given that individual is going to survive. (This is equivalent to function DW which was used in the effectiveness section 2.2.1.1).

```
cos =def cos(cycle,cos_within_first_year,T,cos_after_first_year,cos_sur):
    if cycle<=12:
        return cos_within_first_year
    elif 12<cycle<=T:
        return cos_after_first_year
    else:
        return cos_sur
```

Now we can define COS_ALIVE which will be assigned into “Alive” state as incremental and final costs:

$$COS\_ALIVE = \boldsymbol{Discount}(\boldsymbol{user}(\text{"}cos\text{"}; cycle - 1; cos\_within\_first\_year; T; cos\_after\_first\_year; cos\_sur); discrateC; (cycle - 1)/12) + \boldsymbol{Discount}(\boldsymbol{if}(cycle <= 7; cos\_dir; 0); discrateC; (cycle - 1)/12)$$

2.2.2.2 Costs for state “Dc”

To assign cost to the “Dc” state we again use an approach similar to the effectiveness approach in section 2.2.1.2. We adjust costs assigned by means of COS_ALIVE into “Alive” state for previous 11 cycles.

```
COS_DELTA= Discount(if(cycle-1<=12;max(cos_within_first_year;cos_Dc_last_year);cos_Dc_last_year);discrateC;(cycle-1)/12)+Discount(if(cycle<=7;cos_dir;0);discrateC;(cycle-1)/12)+if(cycle>=15;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-2;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-2)/12)+if(cycle>=16;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-3;cos_within_first_year;T;cos_after_first_year;cos_sur));discrate;(cycle-3)/12)+if(cycle>=17;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-4;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-4)/12)+if(cycle>=18;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-5;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-5)/12)+if(cycle>=19;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-6;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-6)/12)+if(cycle>=20;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-7;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-7)/12)+if(cycle>=21;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-8;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-8)/12)+if(cycle>=22;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-9;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-9)/12)+if(cycle>=23;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-10;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-10)/12)+if(cycle>=24;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-11;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-11)/12)+if(cycle>=25;1;0)*Discount((cos_Dc_last_year-user("cos";cycle-12;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-12)/12)
```

COS_DELTA will be assigned into “Dc” state as an incremental cost.

2.2.2.3 Costs for state “Doc”

Costs for “Doc” state are calculated in the same way as costs in “Dc”, only it is adjusted to $cos\_Doc\_last\_year$:

```
COS _DOC =Discount(if(cycle-1<=12;max(cos_within_first_year;cos_Doc_last_year);cos_Doc_last_year);discrateC;(cycle-1)/12)+Discount(if(cycle<=7;cos_dir;0);discrateC;(cycle-1)/12)+if(cycle>=15;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-2;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-2)/12)+if(cycle>=16;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-3;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-3)/12)+if(cycle>=17;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-4;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-4)/12)+if(cycle>=18;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-5;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-5)/12)+if(cycle>=19;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-6;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-6)/12)+if(cycle>=20;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-7;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-7)/12)+if(cycle>=21;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-8;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-8)/12)+if(cycle>=22;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-9;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-9)/12)+if(cycle>=23;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-10;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-10)/12)+if(cycle>=24;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-11;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-11)/12)+if(cycle>=25;1;0)*Discount((cos_Doc_last_year-user("cos";cycle-12;cos_within_first_year;T;cos_after_first_year;cos_sur));discrateC;(cycle-12)/12)
```

COS_DOC will be assigned into “Doc” state as incremental cost, and to account for people surviving to 110 years also as a final cost.

2.2.3 Modelling heterogeneity via clones

In order to model evaluate cost and effectiveness for any sub-population, we need to run a sequence of cost-effectiveness analyses. For example, if we want to evaluate results for the Māori population versus the non-Māori population, we can create a sequence of analyses for all Māori sub-groups and another sequence of analyses for all non-Māori subgroups. Next, we have to merge results of subgroups and compare with each other.

In this section we will discuss another, more efficient way to incorporate heterogeneity in the model, as well as effectively run cost-effectiveness, sensitivity and uncertainty analysis for any given sub-population. This is achieved by using clones in TreeAge.

Let us again consider the following sub population: ages 50-64 years (50-54, 55-59, and 60-64), with further stratification by ethnicity, gender and deprivation level, and two intervention scenarios:

1) Intervention that reduces cancer mortality rate by 20% with direct costs of $1000 per month (cycle) in first six months;

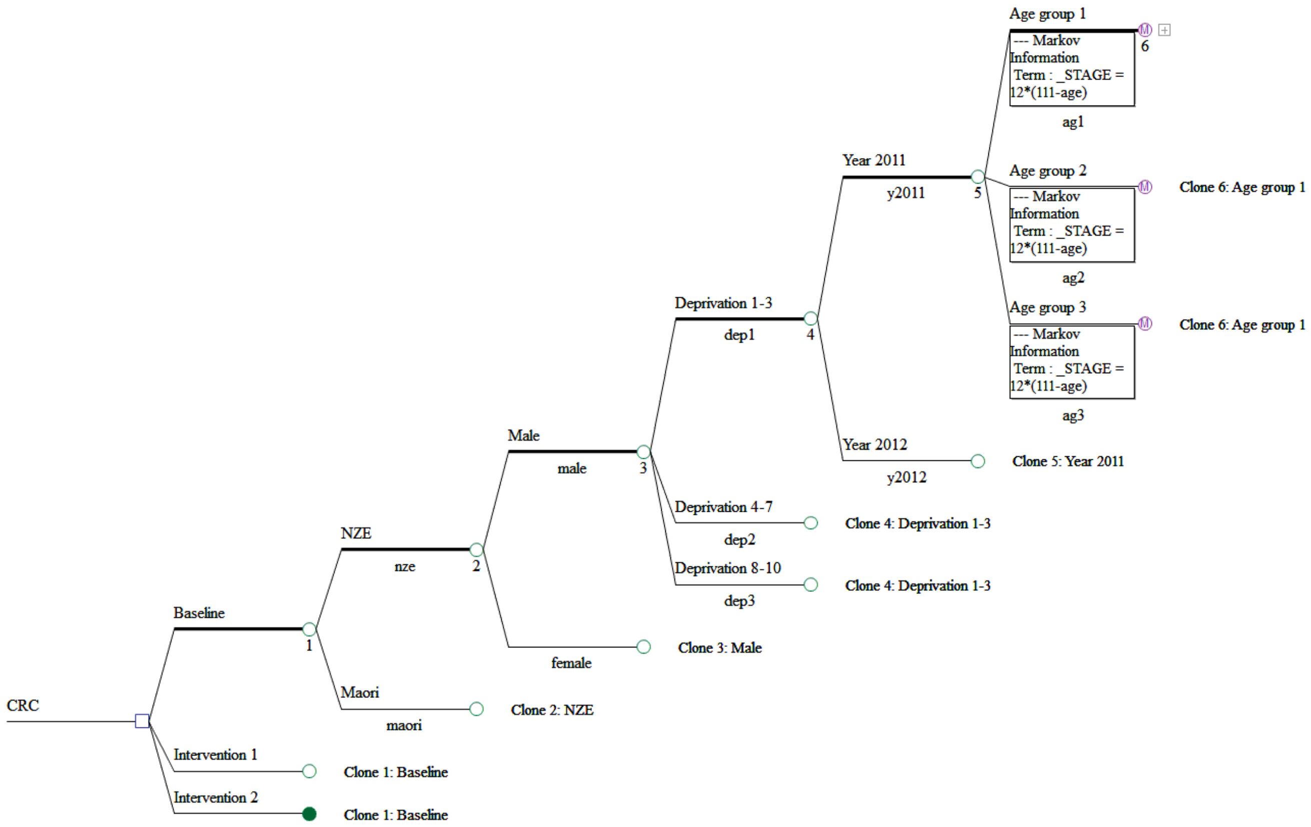
2) Intervention that reduces cancer mortality rate by 40% with direct costs of $5000 per month in first six months

The model starts with a decision node **CRC** with alternative pathways **Baseline**, **Intervention1** or **Intervention2**. The Intervention nodes are clones of the **Baseline** node**,** so that the tree structure developed after the **Baseline** node will be copied into the **Intervention** nodes.

The **Baseline** node is a chance node with two possible paths **NZE** and **Māori,** the latter being a clone of **NZE.** The **NZE** node is a chance node with two possible paths **Male** and **Female –** again the **Male** node is cloned into the **Female** node. The **Male** node is a chance node with three possible paths **Deprivation 1-3**, **Deprivation 4-7** and **Deprivation 8-10**, where **Deprivation 4-7** and **Deprivation 8-10** are clones of **Deprivation 1-3. Deprivation 1-3** is a chance node with several possible pathways (let us for simplicity assume 2 possible alternatives) **Year 2011** and **Year 2012.** Again, **Year 2011** is cloned into **Year 2012**. Finally the **Year 2011** node is a chance node with 3 possible destinations, **Age group 1** (50-54)**, Age group 2** (55-59)**, Age group 3** (60-64)**.**

All these nodes are clones of the node **Age group 1.** The node **Age group 1** is a Markov node with the generic Markov model shown on Figure 4.

Figure 4: CRC model demonstrating clones

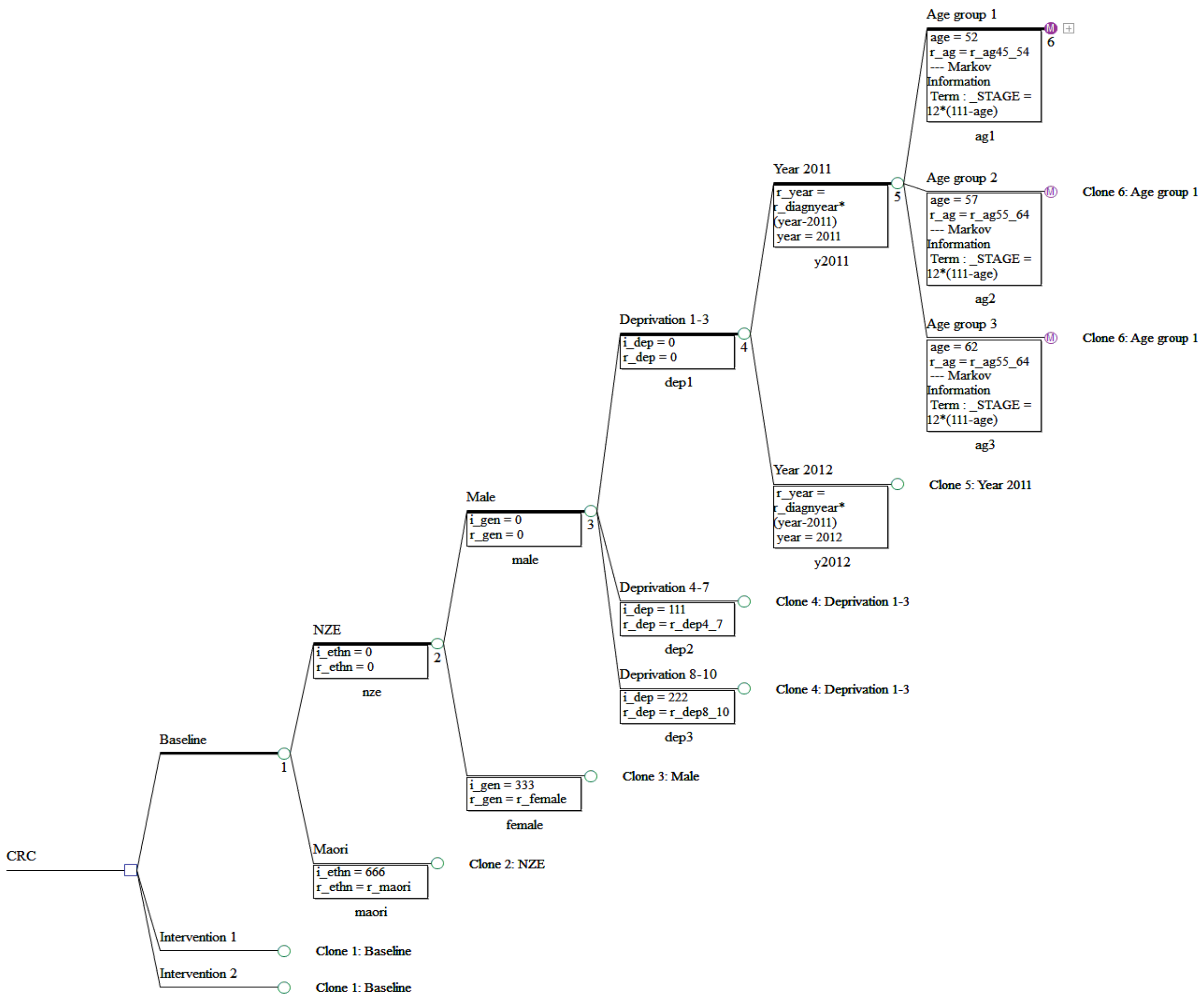


Under this structure it is very easy to run any kind of analysis for any specified strata or group. For example, our strata is NZE/Male/Deprivation 1-3/Year 2011/Age group 1. By setting appropriate transition probabilities (on Figure 1 all transition probabilities are equal to 1, which means that the model includes all strata) we can “switch on” that particular strata and “switch off” all the remaining strata.

For strata described above we will set transition probabilities **Baseline** → **NZE**, **NZE** → **Male**, **Male** → **Deprivation 1-3**, **Deprivation 1-3** → **Year 2011**, **Year 2011**→**Age group 1** equal to 1 and the rest of the transition probabilities equal to 0.

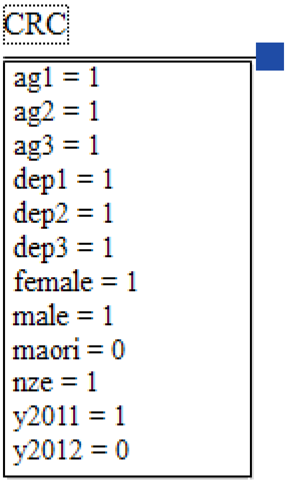
As we see transition probabilities in this tree become model parameters which are used for choosing specific strata or group. This model capability is particularly convenient in the context of equity analysis. We also define all strata specific parameters in the corresponding nodes.

Figure 5: Defining strata specific parameters in nodes



Defining parameters as shown on Figure 5 guarantees that all the strata dependent functions, $\boldsymbol{ind}$, $\boldsymbol{MR}$, $\boldsymbol{EMR}$ defined in the Markov nodes will get assigned appropriate values. Altering strata defining parameters, $\boldsymbol{ag1}$, $\boldsymbol{ag2}$, $\boldsymbol{ag3}$, $\boldsymbol{nze}$, $\boldsymbol{maori}$, $\boldsymbol{male}$, $\boldsymbol{female}$, $\boldsymbol{dep1}$, $\boldsymbol{dep2}$, $\boldsymbol{dep3}$, $\boldsymbol{year2011}$, $\boldsymbol{year2012}$ we can include or exclude any sub-population group into analysis.

Figure 6: Parameter settings to select population groups



CRC

ag1 = 1
ag2 = 1
ag3 = 1
dep1 = 1
dep2 = 1
dep3 = 1
female = 1
male = 1
maori = 0
nze = 1
y2011 = 1
y2012 = 0

For example, Figure 6 shows the following population group: all non-Māori between 50 to 64, all deprivation tertiles, and diagnosed in 2011. To include all Māori into analysis we just change the value of $\boldsymbol{maori}$ from 0 to 1.

2.2.4 Sensitivity and uncertainty analyses

Sensitivity and uncertainty analysis is a tool to investigate uncertainty in results due to uncertainty in the model input parameters. Model input parameter uncertainty can be specified in two ways: 1) Deterministic uncertainty – when we know the range of values the parameter can take, and specific (say) a best, low and high estimate; and 2) probabilistic uncertainty – more informative, when we know the distribution of parameter and specify it accordingly (e.g. as a normal distribution with a given mean and standard deviation).

Sensitivity analysis usually is a scenario analysis performed to obtain the results for different values of model parameters; we specify the range around the parameter and perform cost-effectiveness analysis for different values of parameter. If we want to investigate an impact of only one parameter then we will perform a so called 1-way sensitivity analysis. If we want to investigate an impact of two parameters then we will perform 2-way sensitivity analysis, when the different values of two parameters are obtained from their corresponding ranges.

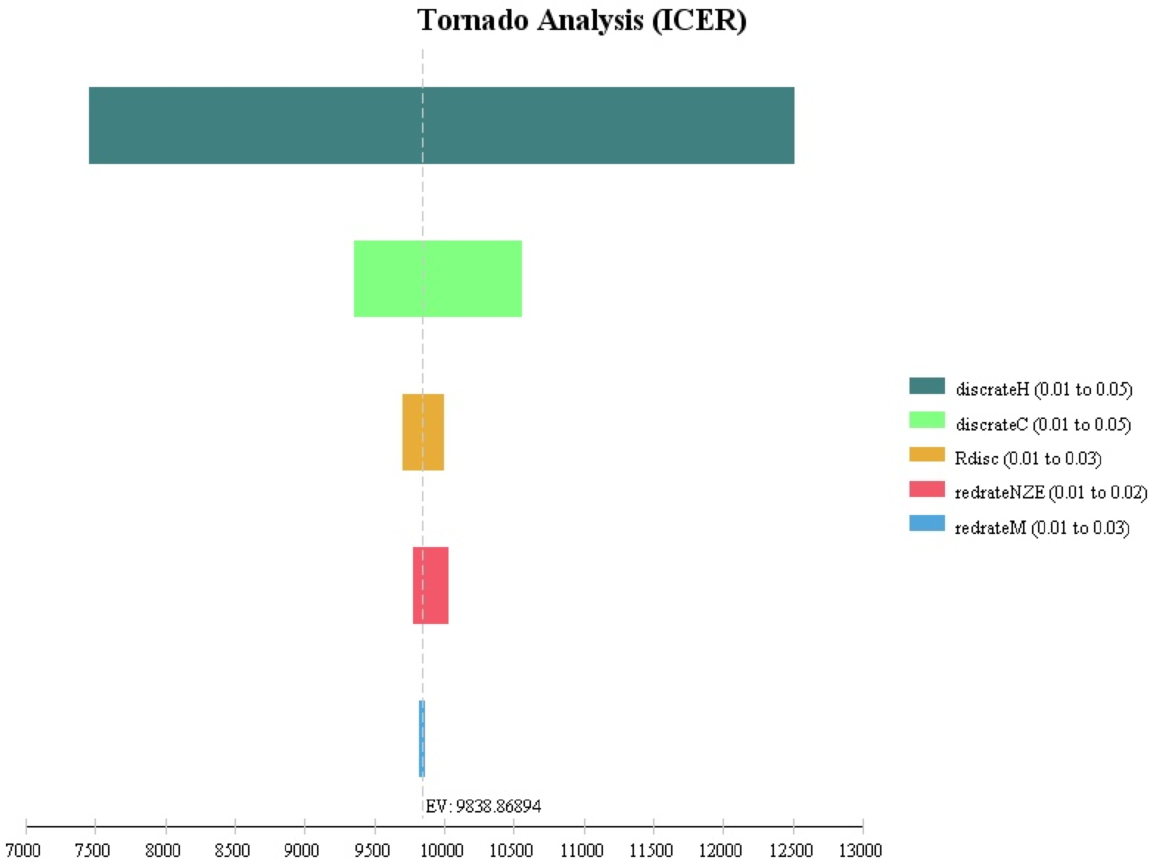
Let us for simplicity consider 1-way sensitivity analysis for parameter $discrateC$- the discount rate for costs. Let us assume that the range for discount rate is (0.01;0.05). We specify the range and number of values from that range we want to apply. TreeAge automatically obtains these values based on uniform partition. For example, if the number of values is equal to 5 then TreeAge will perform the cost-effectiveness analysis for $discrateC = 0.1; 0.2; 0.3; 0.4; 0.5$, as shown in Table 4.

Table 4: Results of 1-Way sensitivity analysis, for all population strata combined

discrateC	**Strategy**	**Cost**	**HALYs**	**CE**	**IncCost**	**IncHALYs**	**ICER**
0.01	Baseline	22148661	5396.9	4104.0	0.0	0.0	0.0
	Intervention 1	27215093	5876.9	4630.8	5066431	480.0	10554.4
	Intervention 2	43307819	6401.8	6765.0	16092726	524.8	30662.7
0.02	Baseline	20215127	5396.9	3745.7	0.0	0.0	0.0
	Intervention 1	25091903	5876.9	4269.6	4876775	480.0	10159.3
	Intervention 2	40946839	6401.8	6396.2	15854935	524.8	30209.6
0.03	Baseline	18656996	5396.9	3457.0	0.0	0.0	0.0
	Intervention 1	23379968	5876.9	3978.3	4722971	480.0	9838.9
	Intervention 2	39036840	6401.8	6097.8	15656872	524.8	29832.2
0.04	Baseline	17382579	5396.9	3220.8	0.0	0.0	0.0
	Intervention 1	21978845	5876.9	3739.8	4596266	480.0	9574.9
	Intervention 2	37467774	6401.8	5852.7	15488928	524.8	29512.2
0.05	Baseline	16325506	5396.9	3025.0	0.0	0.0	0.0
	Intervention 1	20815835	5876.9	3542.0	4490328	480.0	9354.2
	Intervention 2	36159982	6401.8	5648.4	15344147	524.8	29236.4

Another common tool for sensitivity analysis is so called Tornado diagram which represents the impacts of several parameters on results on the same graph. Figure 7 shows a Tornado diagram for Baseline vs Intervention1 ICER for the following parameters: $discrateH$ (discount rate for HALYs), $discrateC$(discount rate for costs), $Rdisc$ (reduction rate in remission state), $redrateNZE$ (reduction rate for NZE), $redrateM$ (reduction rate for NZE). These figures show that the chosen values about the discount rate of HALYs had the largest impact on the ICER, whereas the chosen values for sensitivity analysis about the annual mortality rate reduction in background mortality (i.e. that from lifetables) rate had the smallest impact. It must be emphasised that the outcome of this sensitivity analysis is determined by, first, the parameters we select for sensitivity analysis, and second the degree of parameter uncertainty we specify about each parameter. These considerations will be canvassed in much greater depth in future substantive work.

Figure 7: Tornado sensitivity analysis diagram for Baseline Vs Intervention1 ICER for selected parameters, for all population strata combined



disrateH = the discount rate applied to HALYs; best estimate 3%, low 1% and high 5%.
disrateC = the discount rate applied to costs; best estimate 3%, low 1% and high 5%.
Rdisc = the monthly reduction rate applied to remission disability weight; best estimate 1.84%, low 1% and high 3%.
redrateNZE = the annual reduction rate applied to NZE background mortality rate; best estimate 1.75%, low 1% and high 2%.
redrateM = the annual reduction rate applied to Māori background mortality rate; best estimate 2.25%, low 1% and high 3%.

We now consider some hypothetical probabilistic uncertainty analyses. The parameters that carry probabilistic uncertainty around them are *cos_dir_1*, *cos_dir_2*, *es_1*, *es_2*. These parameters are effect sizes, so we will observe uncertainty only in intervention arms. To create uncertainty, artificially for demonstrative purposes, in the baseline arm let us define distribution $dist\_es\_0{\sim}N(1{,}0.05)$.[4] That is, we assumed that the EMR in the baseline model was variable about its best estimate.

As outlined above, in our case example we assume that Intervention 1 and Intervention 2 reduce excess mortality rate by 20% and 40% respectively. This will be captured by introducing parameter es which takes values es_0, es_1 and es_2 in **Baseline, Intervention 1** and **Intervention 2** nodes respectively. Let us now define two distributions $dist\_es\_1{\sim}N(0.8{,}0.05)$, $dist\_es\_2{\sim}N(0.6{,}0.05)$. Next we define es_1 and es_2 as:

$$es\_0 = dist\_es\_0{\sim}N(1{,}0.05)$$

$$es\_1 = dist\_es\_1{\sim}N(0.8{,}0.05)$$

$$es\_2 = dist\_es\_1{\sim}N(0.6{,}0.05)$$

That is, for intervention 1 the reduction in the EMR has a best estimate of 0.20 (or ratio multiplier of 0.80), with a normal distribution with standard deviation of 0.05.[5] And values are sampled probabilistically from this distribution.

Another effect of intervention is direct costs. Define two distributions $dist\_cos\_dir\_1{\sim}Gamma(m = 1000, SD = 200)$ and $dist\_cos\_dir\_2{\sim}Gamma(m = 5000, SD = 500)$, Then define parameter cos_dir_1 as:

$$cos\_dir\_1 = dist\_cos\_dir\_1$$

and

$$cos\_dir\_2 = dist\_cos\_dir\_2$$

The cos_dir as cos_dir_0, cos_dir_1 and cos_dir_2 were specified in the **Baseline, Intervention 1** and **Intervention 2** nodes, respectively.

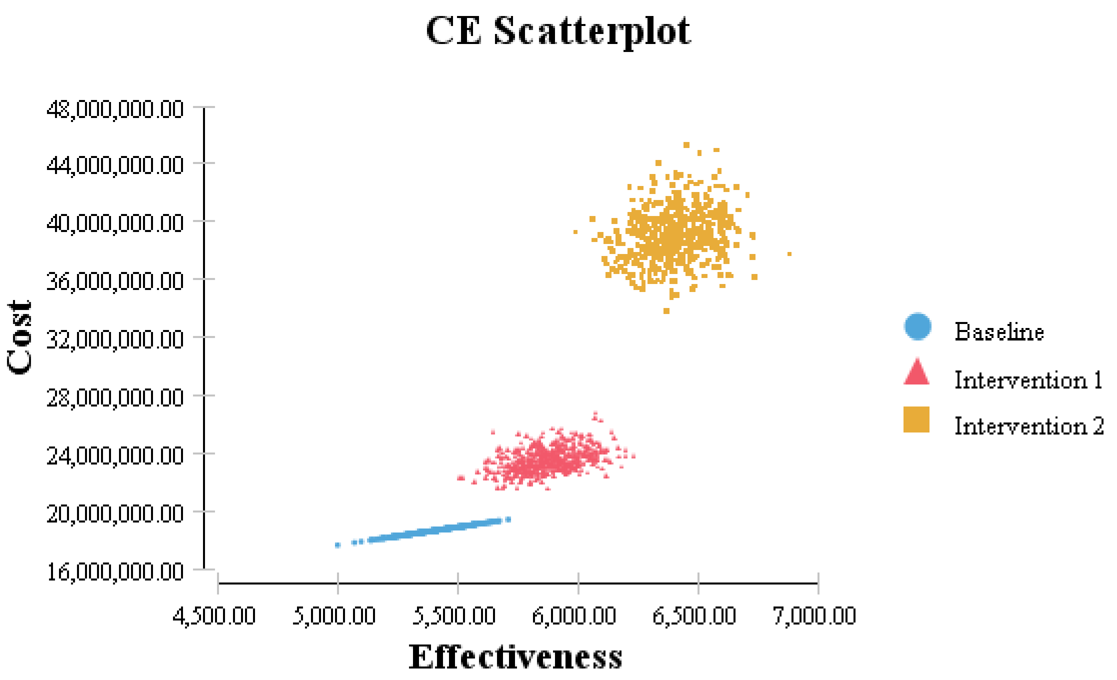
[4] In future evaluations, such uncertainty – if necessary – will probably be captured in the EMR itself, not a contrived effectiveness rate ratio for the comparator.

[5] In future evaluations, a log normal distribution will be used for any ratio measure.

The main conceptual difference between uncertainty analysis and sensitivity analysis is that the uncertainty analysis is evaluated for different values of the input parameters sampled from their respective distributions using a Monte Carlo technique – not discrete or deterministic values. Also, the number of iterations should be reasonably large to fully capture the distributions, meaning that uncertainty analysis is more time consuming than sensitivity analysis. (However, except for complex models this is a trivial issue with modern computers). As discussed above, the model uses six parameters that carry probabilistic uncertainty (distributions) around them. Figure 8 shows the result of probabilistic uncertainty analysis (PSA) for 500 iterations. That is, the cost and effectiveness of baseline, intervention 1 and intervention 2 is evaluated for 500 values randomly sampled from six distributions.

Note that this modeling was done with the following process. At the outer most level, a parameter value was sampled independently for each of the five parameters. Then these sample values were held constant for all cycles of the Markov model (i.e. whatever ratio was sampled for the treatment effectiveness (that was then multiplied by the EMR at each cycle) was held constant across all cycles of the model). Second, all of the heterogeneous strata were run in parallel (with the same sampled input parameters), for all of the baseline, and intervention 1 and 2 clones. This whole process was then repeated another 499 times for a new set of input parameters. (Note that one could conduct this process differently. For example, one could specify correlations between the sampled values of the input parameters. One could also allow the input parameters to be resampled at each cycle).

Figure 8: Cost effectiveness plane scatterplot for hypothetical baseline and interventions, for all population strata combined



Regarding Figure 8, note that there are 500 ICER estimates for each of intervention 1 vs baseline, intervention 2 vs baseline, and intervention 2 vs intervention 1. These could be derived by joining the 'dots' for the same iteration. However, this is clumsy to plot. Rather, one usually presents the median ICER for each relevant comparison, and the 2.5th and 97.5th percentile value. This is shown in Table 5.

Table 6 below shows the incremental costs, HALYs and ICER between intervention 1 and baseline (all three), and between intervention 2 and baseline and intervention 1 (just ICERs). Of note, because of non-linearity in the model the mean ICER ($11,581 for intervention 1 compared to baseline) is not actually the same as the mean incremental cost divided by the mean HALYs ($9781). However, the using the medians in such a way results in a smaller difference ($9880 compared to $9568).

Table 5: Costs, HALYs and ICERs (median and 95% uncertainty intervals) for the analysis shown in Figure 8

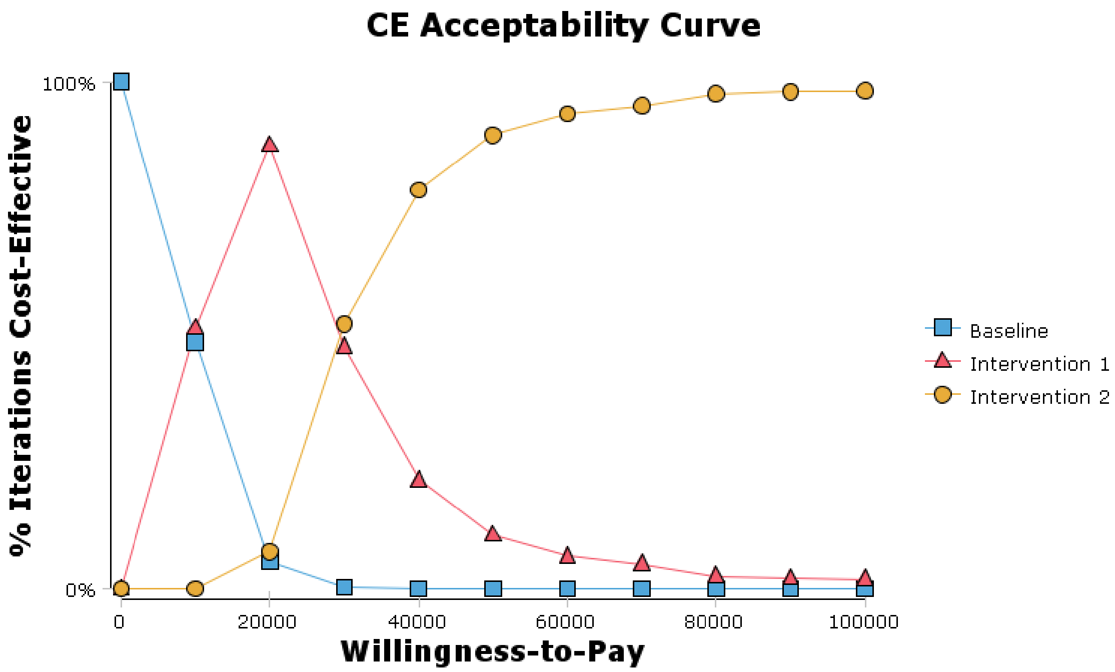
	Baseline		Intervention 1		Intervention2	
	Cost	HALYs	Cost	HALYs	Cost	HALYs
Mean	$18,655,854	5397	$ 23,379,956	5880	$ 38,987,849	6406
SD	$ 289,934	116	$ 807,658	124	$ 1,798,627	132
Minimum	$17,658,714	4999	$ 21,393,193	5512	$ 33,793,942	5992
2.50%	$18,140,236	5191	$ 21,945,693	5643	$ 35,752,906	6149
10%	$18,276,338	5245	$ 22,471,199	5717	$ 36,692,089	6237
Median	$18,647,856	5393	$ 23,300,247	5878	$ 38,901,733	6404
90%	$19,051,285	5555	$ 24,463,992	6047	$ 41,199,459	6588
97.50%	$19,226,381	5625	$ 25,131,421	6116	$ 42,852,355	6636
Maximum	$19,434,646	5709	$ 26,607,023	6234	$ 45,194,434	6881

Table 6: ICERs, IncCost and IncHALYS (median and 95% uncertainty intervals) for the analysis shown in Figure 8

	Intervention 1 vs baseline			Intervention 2 vs baseline	Intervention 2 vs 1
	Incremental Cost	Incremental HALYs	ICER	ICER	ICER
Mean	$ 4,724,101	483	$ 11,581	$ 20,740	$ 36,868
SD	$ 853,422	173	$ 7,645	$ 4,075	$ 44,772
Minimum	$ 2,505,501	33	$ 4,697	$ 11,791	$ 13,786
2.50%	$ 3,085,557	131	$ 6,147	$ 14,954	$ 16,977
10%	$ 3,697,820	263	$ 7,054	$ 16,393	$ 20,647
Median	$ 4,640,395	485	$ 9,880	$ 20,188	$ 29,333
90%	$ 5,823,994	703	$ 16,335	$ 26,067	$ 52,523
97.50%	$ 6,468,849	847	$ 32,539	$ 29,92	$ 78,762
Maximum	$ 7,598,192	953	$ 96,951	$ 44,823	$720,475

In addition to presenting ICERs, it is convenient (and now conventional) to also present results using a net monetary benefit approach, and in particular using cost effectiveness acceptability curves (CEAC). One can see from this figure that for a willingness to pay of less than $10,000 per HALY gained, the baseline scenario has the highest probability of being preferred. For a willingness to pay of $10,000 to $30,000 per HALY gained, intervention 1 is preferred, and above $30,000 intervention 2 is preferred. (Note that it is possible to present such graphs by ethnicity, say, to further understand equity implications of proposed interventions).

Figure 9: Cost Effectiveness acceptability curve for the analysis shown in Figure 8



2.2.5 Equity analysis

Equity often is a significant aspect of cost-effectiveness analysis. How the potential “gain” from interventions, and costs, are distributed across different population groups will be investigated in what we call equity analysis. This section presents a brief first-cut of how such analyses may be undertaken and presented, and explores the impact of various parameters to Māori non-Māori differences.

Table 7: Net costs, HALYs and cost effectiveness for Māori and non-Māori (Expected Values)

		Cost	HALYs	Incremental Costs	Incremental HALYs	ICER
Māori Scen 0	Baseline	$1,318,900	298.3	$0	0	$0
	Intervention 1	$1,740,717	338.9	$421,817	40.5	$10,394
	Intervention 2	$3,110,067	385.2	$1,369,350	46.3	$29,556
NZE	Baseline	$17,338,095	5098.5	$0	0	$0
	Intervention 1	$21,639,249	5538	$4,301,153	439.4	$9,787
	Intervention 2	$35,926,772	6016.5	$14,287,522	478.5	$29,858
Māori Scen 1	Baseline	$1,415,562	341.9	$0	0	$0
	Intervention 1	$1,851,982	389.0	$436,419.	47.0	$9,267
	Intervention 2	$3,239,642	442.8	$1,387,660	53.8	$25,784
Māori Scen 2	Baseline	$1,318,900	338.6	$0	0	$0
	Intervention 1	$1,740,717	384.9	$421,817	46.3	$9,106
	Intervention 2	$3,110,067	437.8	$1,369,350	52.9	$25,880
Māori Scen 3	Baseline	$1,415,562	392.7	$0	0	$0
	Intervention 1	$1,851,982	447.1	$436,419	54.4	$8,018
	Intervention 2	$3,239,642	509.4	$1,387,660	62.2	$22,297

Scenario 0 – Māori have their “own” background mortality and pYLDs
Scenario 1 – Māori have NZE background mortality but “own” pYLDs
Scenario 2 – Māori have NZE pYLDs but “own” background mortality
Scenario 3 – Māori have NZE pYLDs and background mortality

Table 7 above shows a ‘prototypical’ equity analysis. For the above colorectal cancer, the results are presented by Māori and NZE. Not that within each of Māori and NZE the final output is weighted by the actual estimated number of people within each sex by age by deprivation category within each ethnic group – that is, it is representative of these two populations. Thus the first thing we can compare is the total HALYs gained – it is substantially larger for NZE, reflecting the much greater population size (and hence number of incident cases). Likewise the costs. The ICER, however, indirectly takes into account the varying population sizes, and we see that there is little difference – the ICER for intervention 1 versus baseline is better among NZE, but for intervention 2 versus baseline is better for Māori (just).

There are good reasons why the ICER will vary by ethnicity, including ethnic variation in differing excess mortality rates, background mortality rates and pYLDs. So whilst scenario 0 is our best estimate of the actually ICER for both ethnic groups, it is interesting to explore what impact the variation in these parameters has. We use the 1-way sensitivity analysis tool to run four scenarios described above. We introduce new variable $equity$ which takes values from 0 to 3 in such a way that $equity = 0,1,2,3$ in case of Scenario 0, Scenario 1, Scenario 2 and Scenario 3 respectively. Then we simply run 1-way sensitivity analysis on parameter $equity$, specifying lower value equals to 0, high value equals to 3 and . That will return the results for scenarios 0 to 3. Doing so, we see that making the Māori background mortality rate the same as NZE makes the ICERs more favourable for Māori - Māori now have a longer life expectancy, and therefore 'more to gain' by being cured of cancer. Likewise, substituting the NZE pYLD for the Māori pYLD also increases the envelope of possible gains in HALYs for Māori, given the assumptions and structure of the model. These simple comparisons offer insights into possible equity impacts of modeled interventions, and will be pursued elsewhere in publications.

Figure 10: ICER scatterplots for Māori and NZE, (Intervention 1 Vs Baseline) on logarithmic scale (cost on y scale and HALYs on x scale)

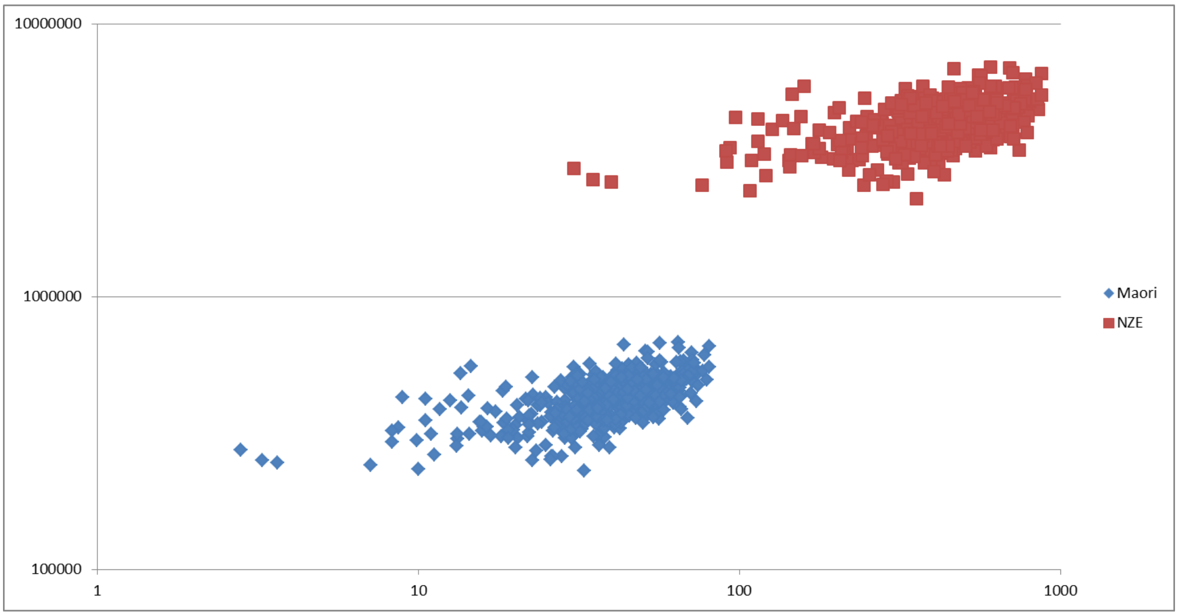


Table 8: ICERs, Incremental Costs and Incremental HALYs (median and 95% uncertainty intervals) for the equity analysis (Intervention 1 Vs Baseline)

	Māori			**NZE**		
	IncCost	**IncHALYs**	**ICER**	**IncCost**	**IncHALYs**	**ICER**
Mean	$ 422,071	40.8	$12,145	$ 4,302,030	442.3	$11,529
SD	$ 76,835	14.7	$ 7,682	$ 776,891	158.6	$ 7,642
Minimum	$ 229,516	2.8	$ 5,204	$ 2,275,985	30.4	$ 4,650
2.50%	$ 276,499	11.1	$ 6,709	$ 2,806,288	119.8	$ 6,096
10%	$ 328,935	22.1	$ 7,574	$ 3,373,234	240.7	$ 7,004
Median	$ 413,733	40.8	$10,440	$ 4,226,505	444.7	$ 9,828
90%	$ 519,984	59.7	$16,886	$ 5,296,649	643.3	$16,291
97.50%	$ 579,046	71.3	$33,217	$ 5,897,245	776.1	$ 32,477
Maximum	$ 678,992	80.6	$ 98,049	$ 6,921,666	872.7	$ 96,850

Note: These results are different from results in **Table 7**, which calculates HALYs and Costs using central estimates of input parameters.

3 More complex Markov models

The above Markov model was a simple three-state model, albeit with highly time and strata dependent parameters and reasonably sophisticated calculations of cost and effectiveness rewards. There are two main ways in which we may extend the Markov models. First, we may increase the number of states. For example, we may add in a remission state, or directly model as a Markov process the disease states shown in Figure 1 (page 3). However, this will often require the specification of transition probabilities that we do not have exact data for – so called model calibration, and the focus of Section 5 below. Second, we may use the same Markov process, but use microsimulation. Here one samples many individual trials within each iteration, which may have advantages when:

- There is marked heterogeneity. For example, by socio-demographic strata as above; here one can just sample individuals based on their probability distribution in the total population. But also heterogeneity for (say) a transition probability that is highly skewed in the population such as growth rates of a tumour; this may confer poor model performance for macro-simulation model that uses population *average* transition probabilities.

- One particularly wants to quantify and understand individual-level stochastic variability (in addition to uncertainty due to input parameters, and heterogeneity).

We will not pursue Markov microsimulation further in this Report, other than to note it is relatively straight forward to swap from macro to micro simulation of a Markov model in TreeAge, and to emphasise that exactly how one conducts the microsimulation will be highly dependent on the intervention, disease model and research question at hand.

4 Discrete Event Simulation

Discrete event simulation (DES) is a form of microsimulation. Compared to Markov modelling, it approaches the issue from a different perspective with *time to event* rather than fixed duration cycle lengths being the key building block for the modelling. Thus, rather than having time dependant transition probabilities from ‘alive’ to ‘death from cancer’ states, one rather specifies a cumulative distribution function of the probability of death from cancer by time. For cancer, this will tend to be an S-shaped curve asymptoting at some probability less than 1.0 (unless everyone eventually dies from the cancer in question). So for a cancer such as colon cancer with approximately 50% survival, the curve will asymptote at about 0.5. One then draws a random number from a uniform distribution from 0 to 1. For any draw above 0.5, the person is a survivor. For a draw close to 0, they would have died within months of diagnosis. For a draw just beneath 0.5, they would have been a person who dies just before the 8 year statistical cure time. And so on. One then also creates cumulative distribution function of the probability of death from all other causes – which will now asymptote at 1.0 as we all eventually die. Now, by drawing two random numbers for the death from cancer and the death from other causes curves, one then just determines which event occurs first (or at all), and this is the first transition in the model.

DES is an extremely flexible modelling strategy. It does require more data on time to event, although often this can be simply derived from empirical observations of mortality rates – which we have. If one wishes, say, to model other transitions (such as time to surgery), then one needs data on the distribution of times in the population of interest – or at least plausible distributions (e.g. gamma for waiting times).

We will not pursue DES further in this Report, other than to note it is possible to undertake in TreeAge or by coding in standalone statistical software. A framework for use of DES in ABC-CBA is being developed in parallel work on care coordinators.

5 Model Calibration

Model calibration involves the identification of input parameter values that produce model output parameters that best predict the observed data (Karnon and Vanni 2011). Models that are not calibrated properly can lead to misleading results and conclusions. Given that much of the ABC-CBA model is built on national data on incidence, survival and stage distribution, much of the models are already 'highly calibrated'. Model calibration becomes an issue when there are additional states added to models for which we do not have directly observed data. An example might be where an intervention prolongs the time in a 'progression-free' state (i.e. equivalent to DT and R states), but does not alter the duration of the 'progressed state' (i.e. equivalent to PT and T states). Thus, we need to extend our basic Markov models (or DES) to accommodate this extra state (or time to event), but ensure that for the baseline model at least the overall survival equates to that we have observed (or predicted) data. This is the main calibration issue we consider in this report.

5.1 Methods to solve unknown transition probabilities: Markov models

It will often be the case that we have input parameter information for some, but not all, transition probabilities in Markov models. In these instances, we need a method to generate a coherent set of model parameters, namely the unknown transition probabilities (or rates). This is more formally called model calibration. In this Section, we present:

- Stylised examples of missing transition probabilities (or time to event functions) that we are likely to encounter in ABC-CBA models, i.e. problem definition.
- A brief overview of model calibration options used for these types of problems in economic decision modelling – and modelling more generally.
- A couple/few worked basic examples using likely preferred options (i.e. Metropolis-Hastings algorithm and Kolmogorov's Forward Equations).
- Options to use in ABC-CBA.

5.2 Problem

We have rich baseline data on incidence rates, excess cancer mortality rates (i.e. survival), and death rates from other causes by age. We also have hypotheses about the structure of likely cancer disease models, e.g.:

- The generic cancer model shown above in Figure 1.

- Information from the literature on average time from diagnosis to disease progression (i.e. progression free survival) and then time in a progressing disease state leading to death from cancer.

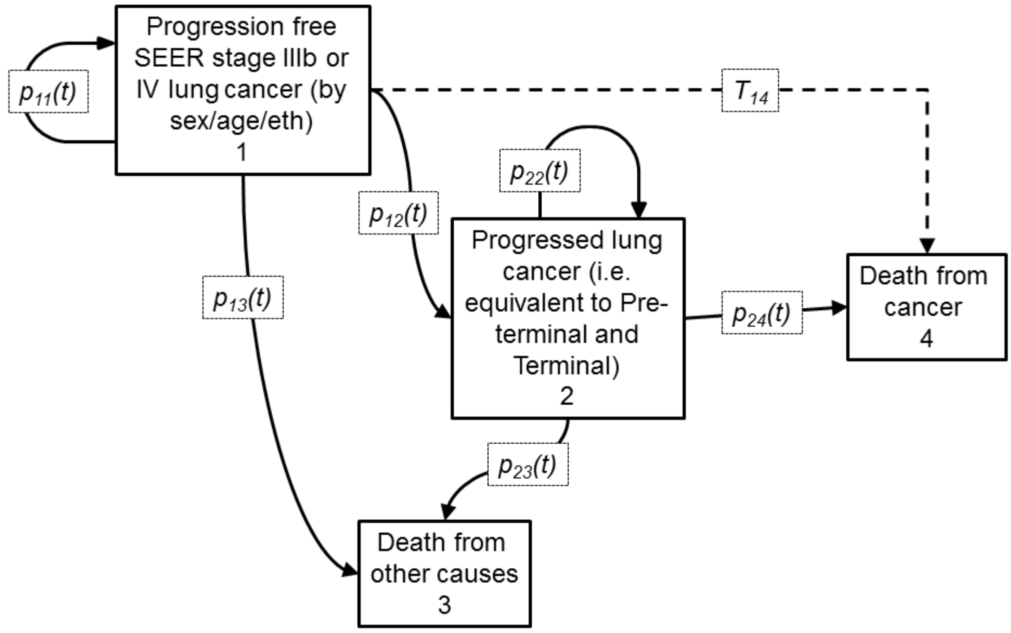
In terms of Markov models, both these instances are more complex than the simple three state model in Section 2. The generic cancer model (Figure 1), if conceptualised as a full Markov model, requires solving for the (highly) time dependent transition probabilities from:

- DT to any of Doc, Dc (i.e. skipping PT and T for rapidly progressing disease),), Remission, PT, T
- R to either DoC or PT
- PT to either DoC or T
- T to either DoC or Dc

Yet, we only have ‘prior’ information on the rate to death from cancer and death from other causes once diagnosed, and hypotheses about model structure.

A simpler example is the creation of a four state system, with the diagnosed state split into ‘progression free [or DT and R]’ and ‘progressing [or PT and T]’ states (Figure 11).

Figure 11: Simplified ABC-CBA four-state model.


Progression free SEER stage IIIb or IV lung cancer (by sex/age/eth)
1
$p_{11}(t)$
T_{14}
$p_{22}(t)$
$p_{12}(t)$
Progressed lung cancer (i.e. equivalent to Pre-terminal and Terminal)
2
$p_{13}(t)$
$p_{24}(t)$
Death from cancer
4
$p_{23}(t)$
Death from other causes
3

States 3 and 4 in Figure 11 are absorbing states. Transition probability p_{13}(t) and p_{23}(t) are known from background mortality rates. The number or proportion of individuals who die from cancer per strata is also known (T_{14}), but we wish to force the transitions state 2 rather than going straight from state 1 to 4. Therefore, p_{12}(t) and p_{24}(t) need to be estimated from (any) existing data or expert opinion, coherently within the envelope or limits set by all other transition probabilities. This is the model calibration task.

5.3 Discrete-time Markov models and calibration

Markov models are used for random processes that evolve through discrete time steps or in a time interval, in which the transition from one state to the next in a discrete state space is governed by a transition probability. The transition to the next state only depends on the current state and none of the previous states, thus making the system “memory-less”.

Discrete-time Markov chain models are ideal when observing patients that progress from one health state to another at fixed points in time. The models are maintained by a transition matrix representing the transition probabilities from one state to another in which the rows and columns indicate the current and future states, respectively. The sum of each row is equal to one. In a discrete-time Markov chain, residence times in specific states obey a geometric distribution, which is known to be the only discrete distribution that exhibits the memory-less property.

Markov models can also be homogeneous or non-homogenous meaning that there is a time dependency in the state transitions. A non-homogeneous model has transition probabilities that are time (e.g. age) dependent and thus more realistic for dealing with observations that can occur at any moment in time and transition probabilities that can vary by time. Although this makes the model more complex and difficult to estimate transition probabilities, there are methods that can handle time dependency by separately solving for age-specific transition probabilities, such as piecewise continuous linear approximations (Whyte, Walsh et al. 2011).

In many cases, we may have access to only the starting state and the final state reached after *n* years, however the path taken to reach this state might be unknown (e.g. example in Figure 11). These data are referred to as *partially observed data*. There are several ways to estimate (or calibrate) the missing transition probabilities or rates, including maximum likelihood and Bayesian approaches.

Calibration approaches in the past have mostly been carried out in the maximum likelihood context, in which, given empirical data, a set of parameters is chosen to be calibrated for a model based on a strategic method for searching the parameter space, a measure of goodness of fit (such as maximizing likelihood function), and criteria for convergence and termination of the search (Vanni, Karnon et al. 2011). Simple calibration approaches involve some form of random grid search of the parameter space such as the Latin hypercube design (Blower, Koelle et al. 2001) but such methods lack in efficiency due to wasteful parameter space searching and inability to distinguish local and global maxima. Although these methods are simpler to comprehend, they immediately become computationally intensive when the parameter space becomes larger in dimension and grid intervals become finer.

There are also a set of simulation based maximum likelihood parameter estimations. (Chia, Salzman et al. 2004) compared three maximum likelihood methods; the Nelder–Mead simplex algorithm (Nelder and Mead 1965), the Kiefer–Wolfowitz algorithm (Kiefer and Wolfowitz 1956), and the Robbins–Monro algorithm (Robbins and Monro 1951), to calibrate their breast cancer tumor growth model. Each method resulted in the estimated parameter values to be within the range of the analytically computed solution, however all methods similarly fail to converge with increasing model complexity. More recently, the EM algorithm has been applied to parameter estimation in multistate progressive models with heterogeneous transition probabilities at an individual or cohort level, however longitudinal data are required and transition probabilities are required to be in one direction only (Chen, Yi et al. 2010).

On the other hand, methods in the Bayesian framework have been seen as a more convenient and natural way to combine evidence from observed data and expert opinion (Spiegelhalter 2004). Bayesian methods have also been adopted in microsimulation studies (Rutter, Miglioretti et al. 2009), which are known to be highly detailed and thus more computationally intense. An example of an efficient Bayesian method is the Metropolis-Hasting algorithm, which is a Markov chain simulation method for sampling from a Bayesian posterior distribution.

Bayesian calibration methods of a natural history model were used to model a state transition structure for progression of colorectal cancer (Whyte, Walsh et al. 2011). Data were available from several sources, including cancer incidence categorized by age and stage, autopsy data on polyp prevalence, cancer and polyp detection rates, results from screening interventions, cancer survival rates per age, cancer stage at diagnosis and time since diagnosis. However not all states were directly observed and so unknown transition probabilities were estimated by the Metropolis-Hastings algorithm. The Metropolis-Hastings algorithm estimated the parameters of the natural history model by sampling multiple sets of parameters from a joint posterior distribution (a distribution that jointly depends on two or more parameters) that is compatible with the observed data. Prior distribution functions for the unknown parameters can be set based on expert opinion. If no prior evidence is available, a non-informative Beta(1,1) distribution can be used.

The Metropolis-Hastings algorithm is known to be computationally more efficient than other methods, such as the Latin Hypercube and Monte Carlo sampling techniques, even in a high dimension parameter space. It also provides a general approach for producing a sequence of samples for a parameter with any probability distribution that may be difficult to sample by classical independence methods by creating a Markov chain in which the current sample depends only on the previous sample. However one must achieve model convergence to ensure a meaningful result, which is true for both Bayesian and maximum likelihood models. A Microsoft Excel macro was supplied by Whyte et al. and currently works with a simple three-state model. One could translate this code to another programming language (e.g.

WinBUGS) and extend it to handle higher state models with two-way transition probabilities. The above example was done in R and the code is available in the Appendix.

5.4 Continuous-time Markov models and calibration

Continuous-time Markov processes handle state transitions that can occur at any time. These are more applicable to real world problems, however in practice the additional model complexity may not be worth the effort. Examples include (Beck and Pauker 1983), (Hazen 2000) and (Hazen and Pellissier 1996).

Continuous Markov processes consist of transition rates (or intensities) which are the number of transitions per unit time or in other words the instantaneous risk of moving from a state to a different state (Jackson 2011). Transitions are represented by a transition intensity matrix (with components a_{ij}) in which the rows and columns indicate the current and future states, respectively, with the requirement that the sum of transition rates in any row of the transition intensity matrix is zero.

In a homogeneous continuous-time Markov process, each time the process enters state *i*, the amount of time (residence or sojourn time) spent in that state before making a transition to a different state is exponentially distributed with mean $-1/a_{ii,}$. The probability of a move from state *i* to state *j* is $-a_{ij}/a_{ii}$.

The calibration of continuous-time Markov processes has been recently done in the maximum likelihood framework with the expectation-maximization algorithm (Chen, Yi et al. 2010). Non-homogenous transition probabilities are represented as functions of time, in which the use of piece-wise constant transition intensities provide a framework for efficiently robust estimates for numerous applications in survival and even history analysis (He and Lawless 2003).

In turn, under the Bayesian context, (Welton and Ades 2005) have shown that the use of Markov chain Monte Carlo methods can be used to achieve unknown transition rates from partially observed data. They make use of the Gibbs sampler, which is a special case of the Metropolis-Hastings algorithm, except each iteration updates one parameter at a time, keeping all others constant, thus moving in the parameter space orthogonally. Furthermore, Kolmogorov's forward equations are an advantageous method to estimate transition probabilities from transition rates. Similar to how (Chen, Yi et al. 2010) have shown, multi-state models can be solved without the limitation of the direction of the transition probabilities being only in one direction. Forward equations are also able to combine information from different studies to solve for unknown transition rates. In addition, the method handles fully and partially observed data, in which solutions are given for two-, three- and four-state models with one- and two-way transitions for both kinds of data. Fully observed data are when we know all the destination states and exact times of all transitions in a continuous time frame. Although this is most ideal for a Markov model, it is usually difficult to achieve in practice, and thus, partially observed data are more common. This is when only a snapshot of state occupancy is known at two points in time.

One crucial advantage to the Bayesian method is the availability of WinBUGS and WBDiff code from (Welton and Ades 2005) and (Welton 2007), which can be used and extended to solve for more complicated models in higher dimensions. The software is also free, very fast and not computationally intensive to run. There are however some drawbacks such as guaranteed model complexity with an increasing number of states and thus even though a solution is achievable it might be computationally intense. Another issue is that all examples are focused on the homogeneous Markov scenario, however extensions to a non-homogeneous problem can be attempted with a piecewise constant approximation within each time or age stratum and estimated separately (e.g. Whyte et al. 2011).

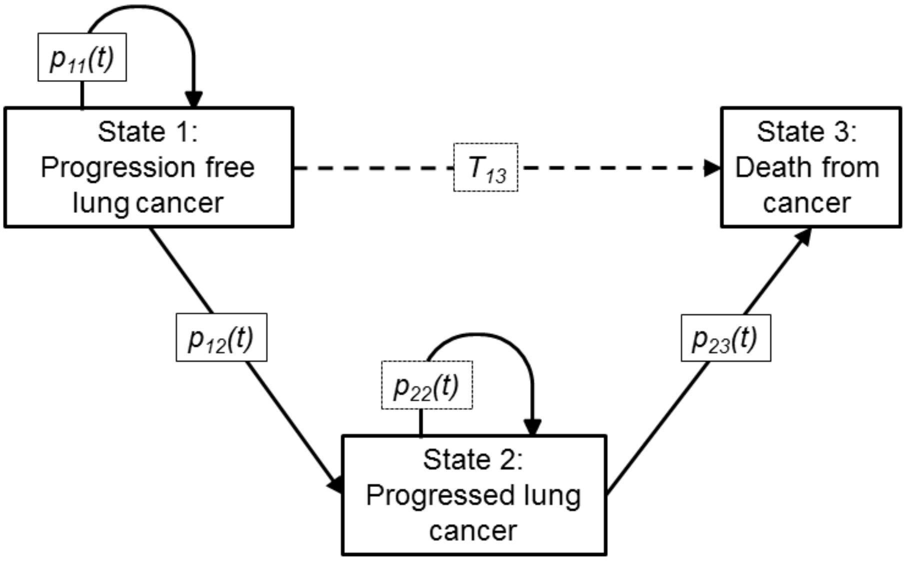
5.5 Methods to solve unknown time to event distributions: Discrete event simulation models

There are also similar parameter calibration and model validation methods for DES, including maximum likelihood (Erenay, Alagoz et al. 2011) and Bayesian approaches (Rios Insua, Ruggeri et al. 2012). They will not be pursued further in this report, but will likely be used in future BODE[3] outputs.

5.6 Worked example: three-state Markov macrosimulation model

We introduce a simple three-state lung cancer model involving progression-free, progressed cancer and death states for patients in a specific age group. We shall use the Metropolis-Hastings algorithm (Whyte, Walsh et al. 2011) to solve for unknown transition probabilities and also Gibbs sampling together with the Kolmogorov equations (Welton and Ades 2005) to solve for unknown transition rates, which will be used to estimate transition probabilities. The three-state model is as follows:

Figure 12: Three-state lung cancer model. All transition probabilities are time (age) dependent.


$p_{11}(t)$
State 1: Progression free lung cancer
T_{13}
State 3: Death from cancer
$p_{12}(t)$
$p_{22}(t)$
$p_{23}(t)$
State 2: Progressed lung cancer

5.6.1 Discrete time Markov models applied to three-state lung cancer model, using Metropolis Hastings and Excel Macro provided by Whyte

As an example, we have the following two data sets of occurrences (Table 8). The first set contains occurrences of progressed lung cancer (state 2) and the total number of individuals in that age group. The second data set is the number of occurrences of individuals who died from cancer (state 1 to state 3). Information from multiple sources can be combined to calibrate the model.

Table 9: Example of data sets for state occurrences of model shows in Figure 11.

	Data set 1: Occurrence of progressed lung cancer		**Data set 2: Occurrences of individuals who died from cancer**	
Age	Occurrences	Total	Occurrences	Total
27	-	-	8	141241
32	-	-	7	124151
37	-	-	15	134112
42	-	-	32	345342
47	-	-	45	636223
52	-	-	109	738712
57	86	1787	211	891411
62	159	2714	239	1010921
67	147	2225	340	1249109
72	77	1026	430	1230112
77	39	486	480	1467090

The model in Figure 12 has a transition probability matrix of:

$$\mathrm{P} = \begin{matrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{matrix} = \begin{matrix} 1 - p_{12} & p_{12} & 0 \\ 0 & 1 - p_{23} & p_{23} \\ 0 & 0 & 1 \end{matrix}$$

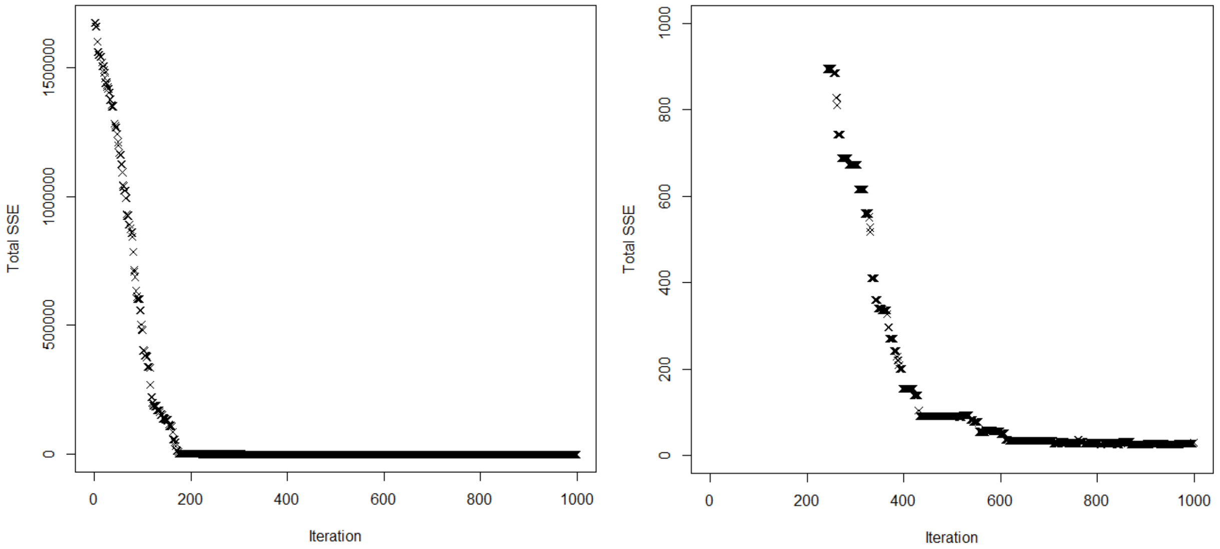
in which the parameters p_{12} and p_{23} have a non-informative prior distribution Beta(1,1).

To start the Markov chain for θ=(p_{12}, p_{23}), we first choose a random set of initial values. Say this is θ_0=(0.02, 0.02). The next step is to choose a candidate parameter set θ’ which is obtained by adding the current set θ_0 to a random sample from a uniform distribution spanning [-ε,ε], say ε =0.001 for both parameters (however ε can take different values for each parameter). The candidate set is accepted based on a criterion comparing the total sum of squared errors (SSE), such that its total SSE is lower than that of the current set. Then the candidate set is renamed as the new current set, and another candidate set is created at random. This is repeated for a high number of iterations (say 5000) and the first 1000 iterations are treated as a burn-in set, resulting in a chain of 4000 candidate sets.

Before calibrating the model, an estimate of the variance of the observed data set is calculated with a small number of iterations (say 1000). A good estimate of the variance is obtained by achieving an acceptance rate between 0.2 and 0.3. The acceptance rate is the number of times a set of candidate values is accepted based on the above criterion divided by the total number of iterations (i.e. 200 to 300 jumps out of the 1000 iterations in a parameter space is considered satisfactory (Whyte, Walsh et al. 2011)).

Convergence of a model can be verified by plotting the SSE versus iteration number. The smaller the SSE, the better calibrated the model is to the data. We can see in Figure 13 that this is achieved at around iteration number 600.

Figure 13: Total sum of squared errors for the first 1000 iterations. The right plot is a cropped version of the left plot



The chain of parameter values can be seen in the Figure 14 below. Both chains start at 0.02, however p_{12} reaches its best estimate quicker (around iteration 200) compared to p_{23}. Plotting the two parameters against each other shows that the Metropolis-Hastings algorithm converges after about 600 iterations (Figure 15).

Figure 14: Markov chains of parameters p_{12} and p_{23}.

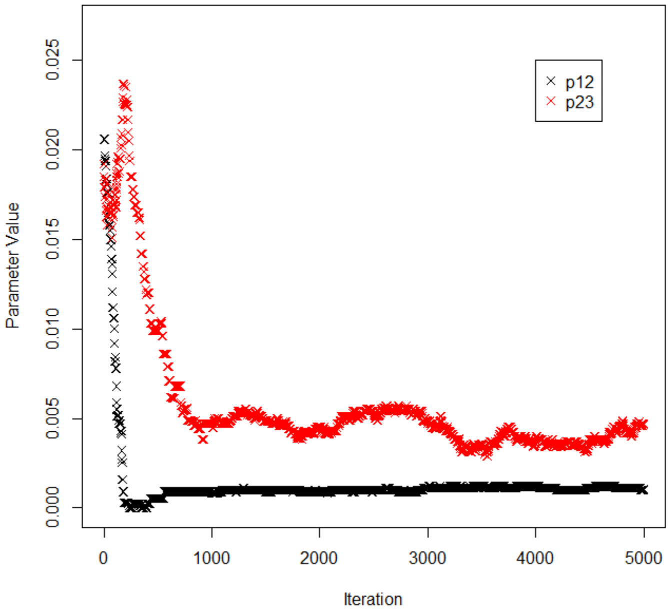
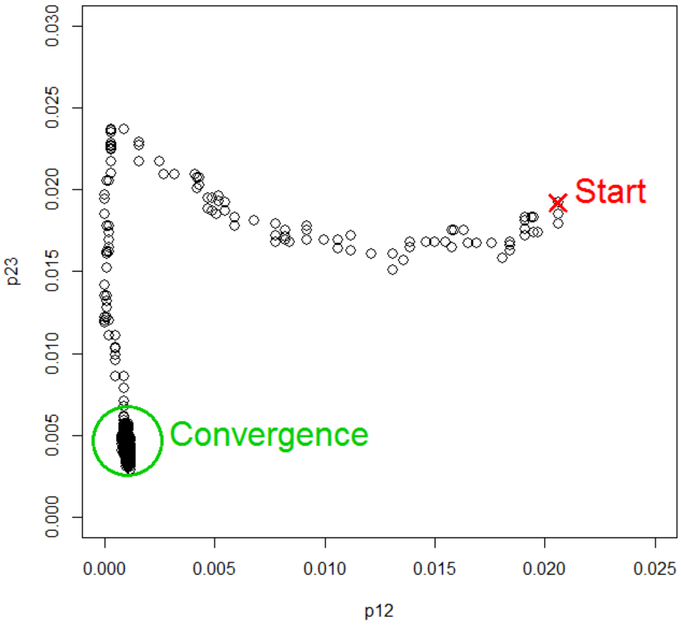


Figure 15: Scatter plot of Markov chains of parameters p12 and p23 both initiated at 0.02 and reached convergence at around 700 iterations.



Finally, we can summarize the Markov chains of the two parameters:

Table 10: Summary of the posterior distributions of transition probabilities p_{12} and p_{23} after 5000 iterations (with 1000 burn-in discarded)

	Minimum	Median	Mean	95% Credible Interval	Maximum
p_{12}	0	0.0010	0.0013	(0.0009, 0.0012)	0.021
p_{23}	0.0029	0.0047	0.0058	(0.0032, 0.0055	0.024

The Excel macro (provided by Sophie Whyte) makes the above modelling with the Metropolis-Hastings algorithm straight forward for this simple example. However, the macro requires some tweaking of the parameters in order to ensure a satisfactory acceptance rate. This step might be tedious depending on the data to be calibrated. Also, in order to extend this method for more complex models, it would be required to recode the macro and create a new spread sheet. A more ideal approach would be to reprogram the method to another language such as R or WinBUGS and allow for options to more complicated models (more states, more time dependency, better selection of prior distributions, etc). Other limitations include the operating system and platform dependency (Windows with MS Office, or Mac with Office for Mac 2011), its inability to run on low-spec computers (e.g. low processing speed or RAM) and it being impossible to interrupt the macro once initiated.

5.6.2 Continuous time Markov models applied to three-state lung cancer model, using Gibbs sampling and Kolmogorov forward equations using WinBUGS

The following example will focus on an alternative case of partially observed data. Say we have the model shown in Figure 11 with the following data for one specific age group/gender/ethnicity at one year after the start of the study:

Table 11: State transition matrix showing the total number of individuals moving from state i to state j after 1 year

	State j		
State i	**1**	**2**	**3**
1	1250	521	0
2	0	3021	710
3	0	0	0

For each state transition matrix, we have a transition intensity matrix of:

$$\mathrm{G} = \begin{matrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{matrix} = \begin{matrix} -g_{12} & g_{12} & 0 \\ 0 & -g_{23} & g_{23} \\ 0 & 0 & 0 \end{matrix}$$

The corresponding transition probability matrix (as a function of time t), P(t) can be obtained from Kolmogorov's forward equation in (Welton 2007) but further simplified to the following:

$\mathrm{P}_{11}\ \mathrm{t}\ = e^{-g_{12}t}$

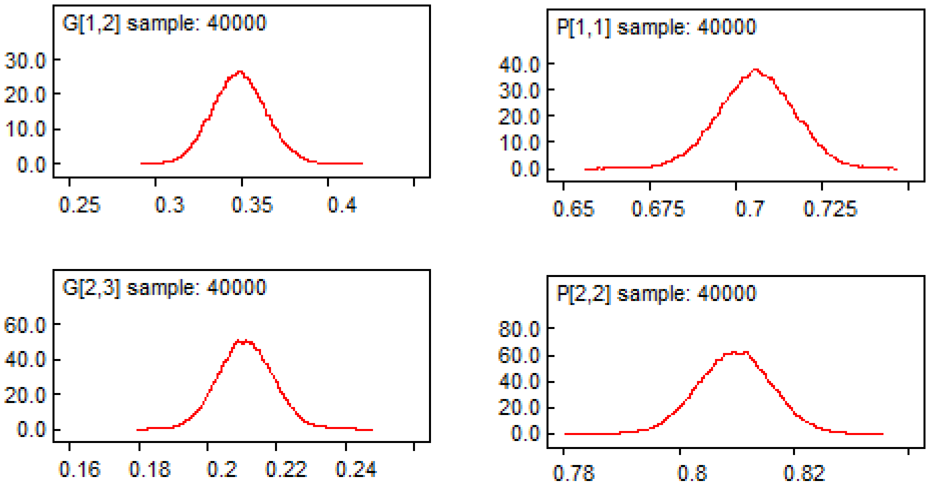
$\mathrm{P}_{12}\ \mathrm{t}\ = 1 - P_{11}(t)$
$\mathrm{P}_{22}\ \mathrm{t}\ = e^{-g_{23}t}$

$P_{23}\ t\ = 1 - P_{22}\ t$

since $P_{13}\ t\ = P_{21}\ t\ = P_{31}\ t\ = P_{32}\ t\ = 0$ and $P_{33}\ t\ = 1$

Calibration of the model was done with 50000 iterations (with burn-in of 10000) and random initialization. We can then obtain density curves for g_{12} and g_{23}, and likewise for P_{11} and P_{22} (Figure 16$_{)}$.

Figure 16: Simulation of transition rates g_{12} and g_{23}, and transition probabilities P_{11} and P_{22} based on 40000 iterations with the Gibbs sampler and Kolmogorov's equations with state-to-state data at 1 year.



Since the calibration of the model is done with respect to a time variable, we can estimate transition probabilities at specific times (Figure 17). Table 12 presents summary statistics of the results from the simulation for transition rates g_{12} and g_{23}, and transition probabilities P_{11} and P_{22} at one year and 0.083 years i.e. one month.

Figure 17: Estimation of transition probabilities P_{11} and P_{22} at 1 month based on 40000 iterations with the Gibbs sampler and Kolmogorov's equations.

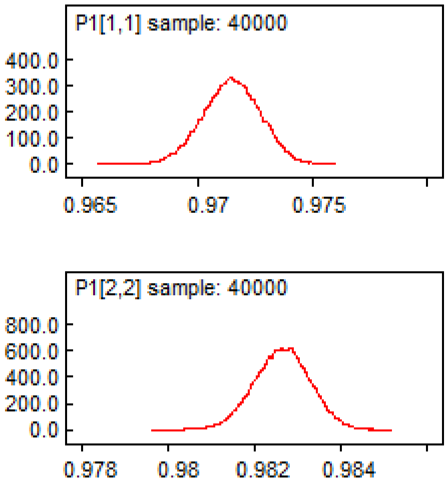


Table 12: Summary statistics of simulation results for transition rates g_{12} and g_{23}, and transition probabilities P_{11} and P_{22} at one year and 0.083 years (one month). This includes the mean estimate, standard deviation, Markov chain error, median and 95% credible interval.

t (year)	Parameter	Mean	SD	MC error	2.5% Q	Median	97.5% Q
	g_{12}	0.349	0.0154	7.43E-05	0.319	0.348	0.380
	g_{23}	0.211	0.00788	3.70E-05	0.196	0.211	0.227
1	P_{11}	0.706	0.0109	5.24E-05	0.684	0.706	0.727
	P_{22}	0.810	0.00638	3.00E-05	0.797	0.8100	0.822
0.083	P_{11}	0.972	0.00125	5.99E-06	0.969	0.972	0.974
	P_{22}	0.983	6.43E-04	3.02E-06	0.981	0.983	0.984

One of the main advantages of this method is that, based on what we know about the states at one point in time, we can calculate transition probabilities for any given time. Another advantage is that the WinBUGS/WBDiff programming language is straight forward and the available code is easily changeable to calibrate more complex models (Welton and Ades 2005). There is less tweaking of parameters compared to the Excel macro for the Metropolis-Hastings algorithm and models can be easily be run with a low-end computer.

5.7 Model calibration: conclusions and options for BODE[3]

Above we have demonstrated model calibration methodologies for a simple three state model, for both transition probabilities and continuous-time transition rates. We used an Excel macro approach, and script written in WinBUGS. We did not, however, specifically address DES model calibration. Also, we analysed only a very simple model (i.e. Figure 12 with transitions to death from cancer, but no transitions to death from other causes as per Figure 11). Our recommendations for future model calibration in BODE^3 are:

- The Excel macro using the Metropolis-Hasting algorithm is a reasonable for understanding the method and visualising the step-by-step process in an easy three-state model with limited
- The macro should be reprogrammed in R and/or WinBUGS for increased flexibility and model complexity, such as a greater number of states (e.g. death from other diseases states, or more disease progression states).
- Extensions to the reprogrammed algorithm can be considered, such as supplying options for prior distribution and handling different input data
- Kolmogorov equation solutions are provided up to a four-state model (Welton 2007) but solving a model greater than four states would be easier to make use of WinBUGS Differential Interface (WBDiff) rather than solving it analytically. This is done by providing a differential equation system that describes the model.

References

1. Atkinson, J., T. Blakely, et al. (2012). Health Tracker Technical Report. Burden of Disease Epidemiology, Equity and Cost-Effectiveness Programme - Technical Report No.11. Wellington, Department of Public Health, University of Otago, Wellington
2. Beck, J. R. and S. G. Pauker (1983). "The Markov Process in Medical Prognosis." Medical Decision Making 3(4): 419-458.
3. Begg, S., T. Vos, et al. (2007). "The burden of disease and injury in Australia 2003." AIHW.
4. Blakely, T., R. Costila, et al. (2010). "The Burden of Cancer: New Zealand 2006."
5. Blakely, T., R. Costilla, et al. (2012). Cancer excess mortality rates over 2006-2026 for ABC-CBA. Burden of Disease Epidemiology, Equity and Cost-Effectiveness Programme - Technical Report No.10. Wellington, Department of Public Health, University of Otago, Wellington
6. Blower, S. M., K. Koelle, et al. (2001). "Live attenuated HIV vaccines: predicting the tradeoff between efficacy and safety." Proceedings of the National Academy of Sciences of the United States of America 98(6): 3618-3623.
7. Chen, B., G. Y. Yi, et al. (2010). "Analysis of interval-censored disease progression data via multi-state models under a nonignorable inspection process." Statistics in Medicine 29(11): 1175-1189.
8. Chia, Y. L., P. Salzman, et al. (2004). "Simulation-based parameter estimation for complex models: a breast cancer natural history modelling illustration." Stat Methods Med Res 13(6): 507-524.
9. Costilla, R., J. Atkinson, et al. (2011). Incorporating Ethnic and Deprivation Variation to Cancer Incidence Estimates over 2006-2026 for ABC-CBA. Burden of Disease Epidemiology, Equity and Cost-Effectiveness Programme - Technical Report No.5. Wellington, Department of Public Health, University of Otago, Wellington
10. Erenay, F. S., O. Alagoz, et al. (2011). "Estimating the Unknown Parameters of the Natural History of Metachronous Colorectal Cancer Using Discrete-Event Simulation." Medical Decision Making 31(4): 611-624.
11. Hazen, G. (2000). "Preference Factoring for Stochastic Trees." Management Science 46(3): 389-403.
12. Hazen, G. and J. Pellissier (1996). "Recursive utility for stochastic trees." Operations Research 44.

13. He, W. and J. F. Lawless (2003). "Flexible Maximum Likelihood Methods for Bivariate Proportional Hazards Models." Biometrics 59(4): 837-848.
14. Jackson, C. (2011). "Multi-State Models for Panel Data: The msm Package for R." Journal of Statistical Software 38(8): 1-28.
15. Karnon, J. and T. Vanni (2011). "Calibrating models in economic evaluation: a comparison of alternative measures of goodness of fit, parameter search strategies and convergence criteria." Pharmacoeconomics 29(1): 51-62.
16. Kiefer, J. and J. Wolfowitz (1956). "Consistency of the Maximum Likelihood Estimator in the Presence of Infinitely Many Incidental Parameters." The Annals of Mathematical Statistics 27(4): 887-906.
17. Kvizhinadze, G. and T. Blakely (2011). Projected NZ Life Tables. Burden of Disease Epidemiology, Equity and Cost-Effectiveness Programme - Technical Report No.4. Wellington, Department of Public Health, University of Otago, Wellington
18. Nelder, J. A. and R. Mead (1965). "A Simplex Method for Function Minimization." The Computer Journal 7(4): 308-313.
19. Rios Insua, D., F. Ruggeri, et al. (2012). Discrete Event Simulation. Bayesian Analysis of Stochastic Process Models, John Wiley & Sons, Ltd: 226-242.
20. Robbins, H. and S. Monro (1951). "A Stochastic Approximation Method." The Annals of Mathematical Statistics 22(3): 400-407.
21. Rutter, C. M., D. L. Miglioretti, et al. (2009). "Bayesian Calibration of Microsimulation Models." Journal of the American Statistical Association 104(488): 1338-1350.
22. Spiegelhalter, D. J. (2004). "Incorporating Bayesian Ideas into Health-Care Evaluation." Statistical Science 19(1): 156-174.
23. Vanni, T., J. Karnon, et al. (2011). "Calibrating models in economic evaluation: a seven-step approach." Pharmacoeconomics 29(1): 35-49.
24. Welton, N. J. (2007). "Solution to Kolmogorov’s equations for some common Markov models ".
25. Welton, N. J. and A. E. Ades (2005). "Estimation of markov chain transition probabilities and rates from fully and partially observed data: uncertainty propagation, evidence synthesis, and model calibration." Medical decision making : an international journal of the Society for Medical Decision Making 25(6): 633-645.

26. Whyte, S., C. Walsh, et al. (2011). "Bayesian calibration of a natural history model with application to a population model for colorectal cancer." Medical decision making : an international journal of the Society for Medical Decision Making 31(4): 625-641.

Appendix One: Metropolis-Hastings Algorithm

The following explains the iterative steps of a Metropolis-Hastings sampler. Suppose we would like to create a Markov chain of parameter sets that obey a target distribution p.

(1) Start with some initial set, x_0

(2) Draw a random candidate set x' which comes from a proposal probability distribution, q, which only depends on x_0. This is usually a distribution that is easy to sample (commonly used choices are normal, binomial, beta or gamma). The closer q is to p, the better the algorithm works.

(3) Compute the ratio

$$R = \frac{\mathrm{p}(\mathrm{x}') \times \mathrm{q}\ \mathrm{x}_0\ \mathrm{x}'}{\mathrm{p}(\mathrm{x}_0) \times \mathrm{q}\ \mathrm{x}'\ \mathrm{x}_0}$$

in which p and q are probability density functions for the target and proposal distributions, respectively.

(4) Define an acceptance probability as the minimum of R and 1.

(5) Let u be randomly selected from a uniform distribution ranging from 0 to 1.

(6) If u < R, accept the candidate set x' and assign x_1 = x'. Otherwise, x_1=x_0.

(7) Return to step (2) and repeat for x_j, for which j=2,..,M.

(8) Discard a burn-in set j=0,..,B for which B<M.

(9) (x_{b+1}, x_{b+2}, .., x_B) is the Metropolis-Hastings sample of parameter sets that comes from the posterior distribution p(x).

Appendix Two: R Code for Metropolis-Hastings Algorithm

```
# Define true values of model parameters
trueA <- 5
trueB <- 1
trueSd <- 10
sampleSize <- 101

# Create independent values x
x <- -50:50

# Create dependent values according to ax + b + N(0,sd)
y <- trueB + trueA * x + rnorm(n=sampleSize, mean=0, sd=trueSd)

# Plot the sample data
plot(x,y, main="")

# Specify prior (uniform) distributions for parameters
a.prior <- c(2,10)
b.prior <- c(-10,10)
s.prior <- c(5,20)

logPrior <- function(a, b, std){
  tmp <- dunif(x=a, a.prior[1], a.prior[2], log=T) +
         dunif(x=b, b.prior[1], b.prior[2], log=T) +
         dunif(x=std, s.prior[1], s.prior[2], log=T)
  return(tmp)
}

# Specify likelihood function

logLik <- function(a, b, std){
          out <- a*x + b
          if(std<0) std = 0
          tmp <- sum(dnorm(x=y, mean=out, sd=std, log=T))
          return(tmp)
        }

# Plot the likelihood profile (for a only)
LL <- function(x){return(logLik(x, trueB, trueSd))}
LLVal <- lapply(seq(2, 10, by=.05), LL)
plot (seq(2, 10, by=.05), LLVal, type="l", xlab = "a",
                ylab = "Log Likelihood")

## Metropolis-Hastings algorithm to sample from posterior density ##

# Initial settings for algorithm
chain.length <- 50000
proposal.width <- c(0.078,0.5,0.5)
```

```
# Initial values (a, b, s), its logLik value and whether it is
# accepted based on the decision criterion below
last <- c(mean(a.prior), mean(b.prior), mean(s.prior),
                        logLik(mean(a.prior),mean(b.prior),mean(s.prior)), 1)

# Set up storage output for candidate values
current <- rep(NA, 5)
chain <- array(NA, dim=c(chain.length, 5))

# Assign first candidate as initial values
chain[1,] <- last

# Iterate for entire chain.length
for (i in 1:(chain.length-1)){
  # Propose new values and calculate LL
  current[1:3] <- rnorm(3, mean=last[1:3], sd = proposal.width)
    current[4] <- logLik(current[1],current[2],current[3]) +
                                     logPrior(current[1],current[2],current[3])
  current[5] <- 1

  # Calculate decision criterion
  R <- exp(current[4] - last[4])

    # Decide whether to accept or not
    alpha <- runif(1, 0, 1)


    if (alpha <= R) {
            chain[i+1,] <- current
            last <- current
            }
  else {
     last[5] <- 0
     chain[i+1,] <- last
            }
}

# Calculate acceptance rate
burnIn <- 10000
AR <- mean(chain[-(1:burnIn),5])

# Summarize results, and calculate 95% credible interval (for a only)
summary(chain[-(1:burnIn),1])
quantile(chain[-(1:burnIn),1],probs=c(0.025,0.975))

# Plot the results (for a only)
par(mfrow = c(2,1))
hist(chain[-(1:burnIn),1], nclass=30, , main="Posterior of a",
            xlab="True value = red line" , xlim=a.prior)
abline(v = mean(chain[-(1:burnIn),1]))
abline(v = trueA, col="red" )
plot(chain[-(1:burnIn),1], type = "l", xlab="True value = red line" ,
        ylab="", main = "Chain values of a", )
abline(h = trueA, col="red" )

# For comparison with linear regression:
summary(lm(y~x))
confint(lm(y~x))
```