

# 3-state cohort state transition model in R

The DARTH workgroup

Developed by the Decision Analysis in R for Technologies in Health (DARTH) workgroup:

Fernando Alarid-Escudero, PhD (1)

Eva A. Enns, MS, PhD (2)

M.G. Myriam Hunink, MD, PhD (3,4)

Hawre J. Jalal, MD, PhD (5)

Eline M. Krijkamp, MSc (3)

Petros Pechlivanoglou, PhD (6,7)

Alan Yang, MSc (7)

In collaboration of:

1. Stanford University, Stanford, CA, USA
2. University of Minnesota School of Public Health, Minneapolis, MN, USA
3. Erasmus MC, Rotterdam, The Netherlands
4. Harvard T.H. Chan School of Public Health, Boston, USA
5. University of Ottawa, Ottawa, ON, Canada
6. University of Toronto, Toronto ON, Canada
7. The Hospital for Sick Children, Toronto ON, Canada

Please cite our publications when using this code:

- Alarid-Escudero F, Krijkamp EM, Enns EA, Yang A, Hunink MGM Pechlivanoglou P, Jalal H. An Introductory Tutorial on Cohort State-Transition Models in R Using a Cost-Effectiveness Analysis Example. *Medical Decision Making*, 2022 (Epub). <https://doi.org/10.1177/0272989X221103163>
- Alarid-Escudero F, Krijkamp EM, Enns EA, Yang A, Hunink MGM Pechlivanoglou P, Jalal H. A Tutorial on Time-Dependent Cohort State-Transition Models in R using a Cost-Effectiveness Analysis Example. *Medical Decision Making*, 2022 (Epub). <https://doi.org/10.1177/0272989X221121747>
- Jalal H, Pechlivanoglou P, Krijkamp E, Alarid-Escudero F, Enns E, Hunink MG. An Overview of R in Health Decision Sciences. *Med Decis Making*. 2017; 37(3): 735-746. <https://journals.sagepub.com/doi/abs/10.1177/0272989X16686559>
- Krijkamp EM, Alarid-Escudero F, Enns EA, Jalal HJ, Hunink MGM, Pechlivanoglou P. Microsimulation modeling for health decision sciences using R: A tutorial. *Med Decis Making*. 2018;38(3):400–22. <https://journals.sagepub.com/doi/abs/10.1177/0272989X18754513>
- Krijkamp EM, Alarid-Escudero F, Enns E, Pechlivanoglou P, Hunink MM, Jalal H. A Multidimensional Array Representation of State-Transition Model Dynamics. *Med Decis Mak*. 2020;40(2):242-248. <https://doi.org/10.1177/0272989X19893973>

Copyright 2017, THE HOSPITAL FOR SICK CHILDREN AND THE COLLABORATING INSTITUTIONS. All rights reserved in Canada, the United States and worldwide. Copyright, trademarks, trade names and any and all associated intellectual property are exclusively owned by THE HOSPITAL FOR Sick CHILDREN and the collaborating institutions. These materials may be used, reproduced, modified, distributed and adapted with proper attribution.

Change `eval` to `TRUE` if you want to knit this document.

```
rm(list = ls())      # clear memory (removes all the variables from the workspace)
```

## 01 Load packages

```
# use this package to conveniently install other packages
if (!require('pacman')) install.packages('pacman'); library(pacman)

# load (install if required) packages from CRAN
p_load("devtools", "diagram", "dampack", "scales")

# load (install if required) packages from GitHub
# install_github("DARTH-git/darthtools", force = TRUE) #Uncomment if there is a newer version
p_load_gh("DARTH-git/darthtools")
```

## 02 Load functions

```
# all functions are in the darthtools package
```

## 03 Model parameters

### 03.1 Define model input parameters

```
### General setup
n_time_horizon_yr <- 60                # time horizon (in years)
cycle_length      <- 1                # cycle length in years (use 1/12 for monthly)
n_cycles          <- n_time_horizon_yr / cycle_length # number of cycles
v_names_cycles    <- paste("cycle", 0:n_cycles)      # cycle names
v_names_states    <- c("Healthy", "Sick", "Dead")    # state names
n_states          <- length(v_names_states)         # number of health states

### Discounting rate
d_c <- 0.03 # annual discount rate for costs
d_e <- 0.03 # annual discount rate for QALYs

### Strategies
v_names_str <- c("Standard of Care",      # store the strategy names
                "Treatment A",
                "Treatment B")
n_str       <- length(v_names_str)       # number of strategies

### Within-cycle correction (WCC) using Simpson's 1/3 rule
v_wcc <- gen_wcc(n_cycles = n_cycles, method = "Simpson1/3")
```

```

### Transition rates
p_HD_yr      <- 0.01 # annual probability of dying when healthy
p_SD_yr      <- 0.10 # annual probability of dying when sick
# Annual probability of becoming sick from Healthy, conditional on surviving cycle
p_HS_yr_SoC  <- 0.05 # under standard of care
p_HS_yr_trtA <- 0.04 # under treatment A
p_HS_yr_trtB <- 0.02 # under treatment B

### State rewards
#### Costs
c_H_yr      <- 400 # cost of one year in healthy state
c_S_yr      <- 1000 # cost of one year in sick state
c_D_yr      <- 0 # cost of one year in dead state
c_trtA_yr   <- 800 # cost of treatment A (per year) in healthy state
c_trtB_yr   <- 1500 # cost of treatment B (per year) in healthy state
#### Utilities
u_H         <- 1 # utility when healthy
u_S         <- 0.5 # utility when sick
u_D         <- 0 # utility when dead

```

### 03.2 Calculate internal model parameters

```

### Cycle-specific discount weight for costs and effects
v_dwc <- 1 / ((1 + (d_e * cycle_length)) ^ (0:n_cycles))
v_dwe <- 1 / ((1 + (d_c * cycle_length)) ^ (0:n_cycles))

### Calculate cycle-specific transition probabilities
p_HD <- 1 - exp(-(-log(1 - p_HD_yr) * cycle_length)) # probability of dying from Healthy
p_SD <- 1 - exp(-(-log(1 - p_SD_yr) * cycle_length)) # probability of dying from Sick
# probability of becoming sick from Healthy
p_HS_SoC <- 1 - exp(-(-log(1 - p_HS_yr_SoC) * cycle_length)) # Standard of Care
p_HS_trtA <- 1 - exp(-(-log(1 - p_HS_yr_trtA) * cycle_length)) # Treatment A
p_HS_trtB <- 1 - exp(-(-log(1 - p_HS_yr_trtB) * cycle_length)) # Treatment B

### Calculate cycle-specific state rewards
#### Costs
v_c_SoC <- c(c_H_yr, c_S_yr, c_D_yr) * cycle_length # Standard of Care
v_c_trtA <- c(c_H_yr + c_trtA_yr, c_S_yr, c_D_yr) * cycle_length # Treatment A
v_c_trtB <- c(c_H_yr + c_trtB_yr, c_S_yr, c_D_yr) * cycle_length # Treatment B
#### QALYs
v_q_SoC <- c(u_H, u_S, u_D) * cycle_length # Standard of Care
v_q_trtA <- v_q_trtB <- v_q_SoC # Treatments A and B have same utilities as SoC

```

## 04 Construct state-transition models (one for each strategy)

```

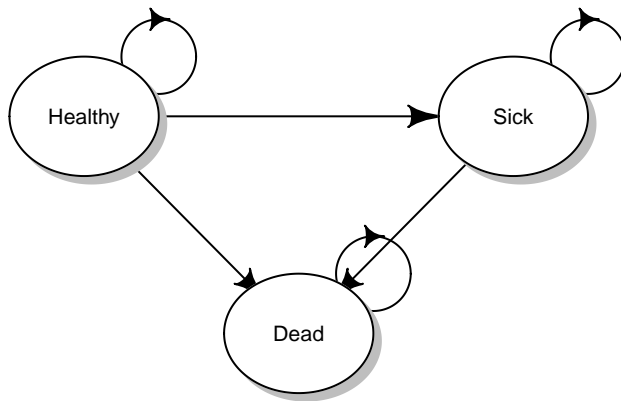
m_P_diag <- matrix(0, nrow = n_states, ncol = n_states, dimnames = list(v_names_states, v_names_states))
m_P_diag["Healthy", "Sick"] = ""
m_P_diag["Healthy", "Dead"] = ""

```

```

m_P_diag["Healthy", "Healthy" ] = ""
m_P_diag["Sick"    , "Dead"   ] = ""
m_P_diag["Sick"    , "Sick"   ] = ""
m_P_diag["Dead"    , "Dead"   ] = ""
layout.fig <- c(2, 1)
plotmat(t(m_P_diag), t(layout.fig), self.cex = 0.5, curve = 0, arr.pos = 0.8,
        latex = T, arr.type = "curved", relsize = 0.85, box.prop = 0.8,
        cex = 0.8, box.cex = 0.7, lwd = 1)

```



## 04.1 Initial state vector

```

# All starting healthy
v_s_init <- c("Healthy" = 1, "Sick" = 0, "Dead" = 0)
v_s_init

```

```

## Healthy    Sick    Dead
##          1      0      0

```

## 04.2 Initialize cohort traces

```

### Initialize cohort trace for SoC
m_M_SoC <- matrix(0,
                  nrow = (n_cycles + 1), ncol = n_states,
                  dimnames = list(v_names_cycles, v_names_states))
# Store the initial state vector in the first row of the cohort trace
m_M_SoC[1, ] <- v_s_init

### Initialize cohort traces for treatments A and B
# Structure and initial states are the same as for SoC
m_M_trtA <- m_M_trtB <- m_M_SoC

```

## 04.3 Create transition probability matrices

```

## Create transition probability matrices for strategy SoC
### Initialize transition probability matrix
# All transitions to a non-death state are assumed to be conditional on survival
m_P_SoC <- matrix(0,
                  nrow = n_states, ncol = n_states,
                  dimnames = list(v_names_states, v_names_states))

### Fill in matrix
# from Healthy
m_P_SoC["Healthy", "Healthy"] <- (1 - p_HD) * (1 - p_HS_SoC)
m_P_SoC["Healthy", "Sick"] <- (1 - p_HD) * p_HS_SoC
m_P_SoC["Healthy", "Dead"] <- p_HD
# from Sick
m_P_SoC["Sick", "Sick"] <- 1 - p_SD
m_P_SoC["Sick", "Dead"] <- p_SD
# from Dead
m_P_SoC["Dead", "Dead"] <- 1

## Treatment A
# Start with same matrix as SoC, but replace parameters that differ for trtA
m_P_trtA <- m_P_SoC
m_P_trtA["Healthy", "Healthy"] <- (1 - p_HD) * (1 - p_HS_trtA)
m_P_trtA["Healthy", "Sick"] <- (1 - p_HD) * p_HS_trtA

## Treatment B
# Start with same matrix as SoC, but replace parameters that differ for trtB
m_P_trtB <- m_P_SoC
m_P_trtB["Healthy", "Healthy"] <- (1 - p_HD) * (1 - p_HS_trtB)
m_P_trtB["Healthy", "Sick"] <- (1 - p_HD) * p_HS_trtB

## Check if transition probability matrices are valid
### Check that transition probabilities are in [0, 1]
check_transition_probability(m_P_SoC, verbose = TRUE)

## [1] "Valid transition probabilities"

check_transition_probability(m_P_trtA, verbose = TRUE)

## [1] "Valid transition probabilities"

check_transition_probability(m_P_trtB, verbose = TRUE)

## [1] "Valid transition probabilities"

### Check that all rows sum to 1
check_sum_of_transition_array(m_P_SoC, n_states = n_states, n_cycles = n_cycles, verbose = TRUE)

## [1] "This is a valid transition matrix"

```

```

check_sum_of_transition_array(m_P_trtA, n_states = n_states, n_cycles = n_cycles, verbose = TRUE)

## [1] "This is a valid transition matrix"

check_sum_of_transition_array(m_P_trtB, n_states = n_states, n_cycles = n_cycles, verbose = TRUE)

## [1] "This is a valid transition matrix"

```

## 05 Run cohort state transition model

```

## Loop over time
# Calculating cohort state based on previous state and transition matrix
for (t in 1:n_cycles){
  # For SoC
  m_M_SoC [t + 1, ] <- m_M_SoC [t, ] %*% m_P_SoC
  # For treatment A
  m_M_trtA[t + 1, ] <- m_M_trtA[t, ] %*% m_P_trtA
  # For treatment B
  m_M_trtB[t + 1, ] <- m_M_trtB[t, ] %*% m_P_trtB
}

```

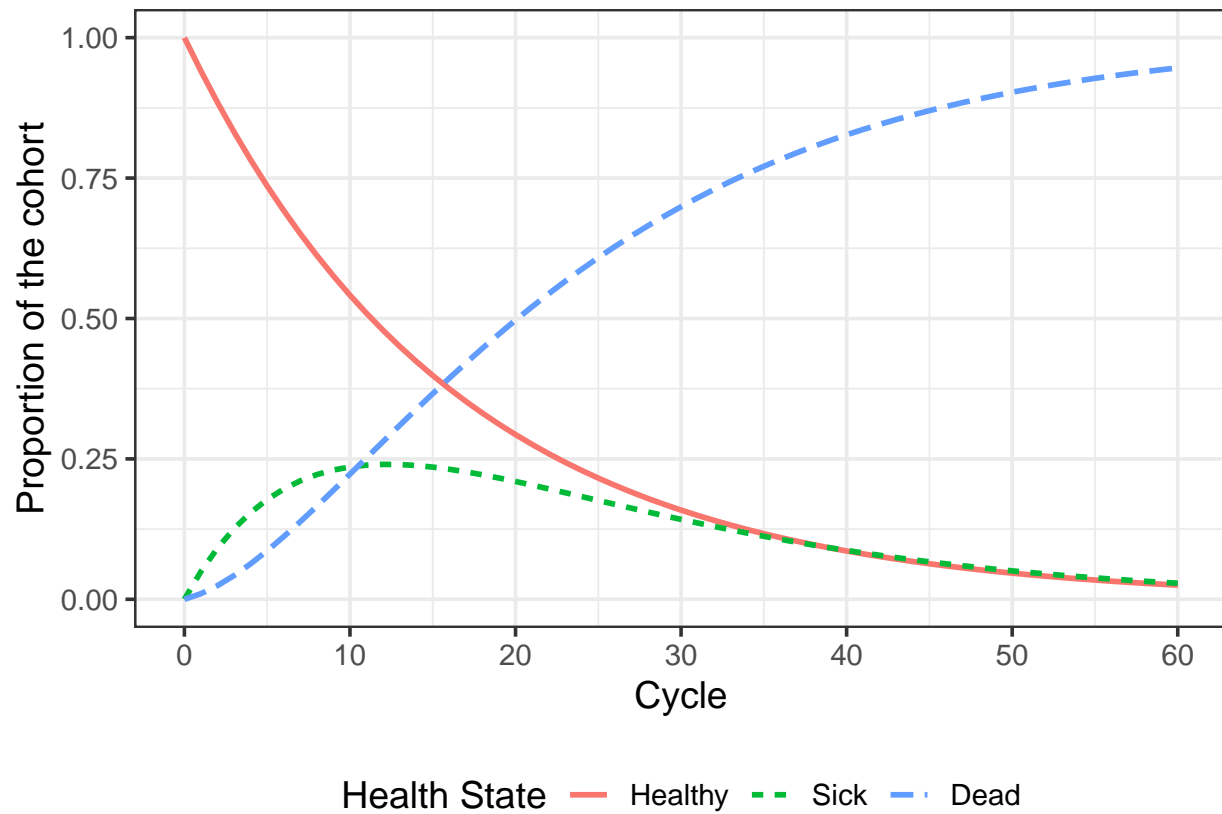
## 06 Plot Outputs

### 06.1 Plot the cohort trace for strategies SoC

```

plot_trace(m_M_SoC)

```



## 06.2 Overall Survival (OS)

Print the overall survival for the Standard of Care

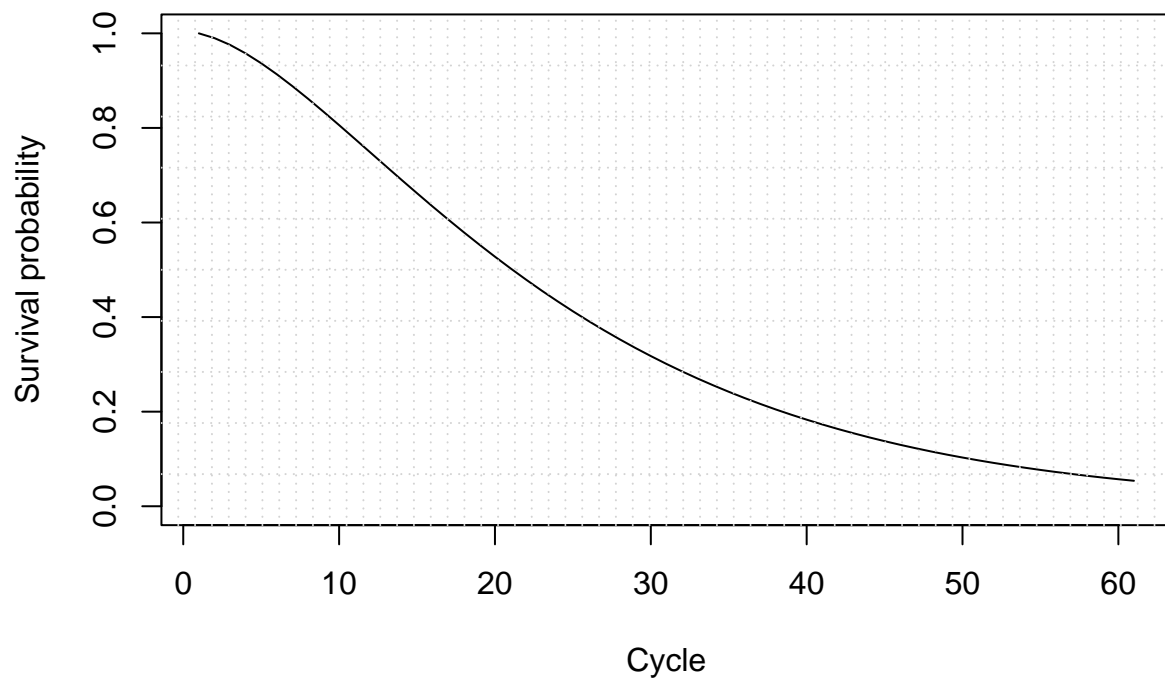
```
v_os_SoC <- 1 - m_M_SoC[, "Dead"] # calculate the overall survival (OS) probability
v_os_SoC <- rowSums(m_M_SoC[, 1:2]) # alternative way of calculating the OS probability

plot(v_os_SoC, type = 'l',
     ylim = c(0, 1),
     ylab = "Survival probability",
     xlab = "Cycle",
     main = "Overall Survival") # create a simple plot showing the OS

# add grid
grid(nx = n_cycles, ny = 10, col = "lightgray", lty = "dotted", lwd = par("lwd"),
     equilogs = TRUE)
```



## Overall Survival



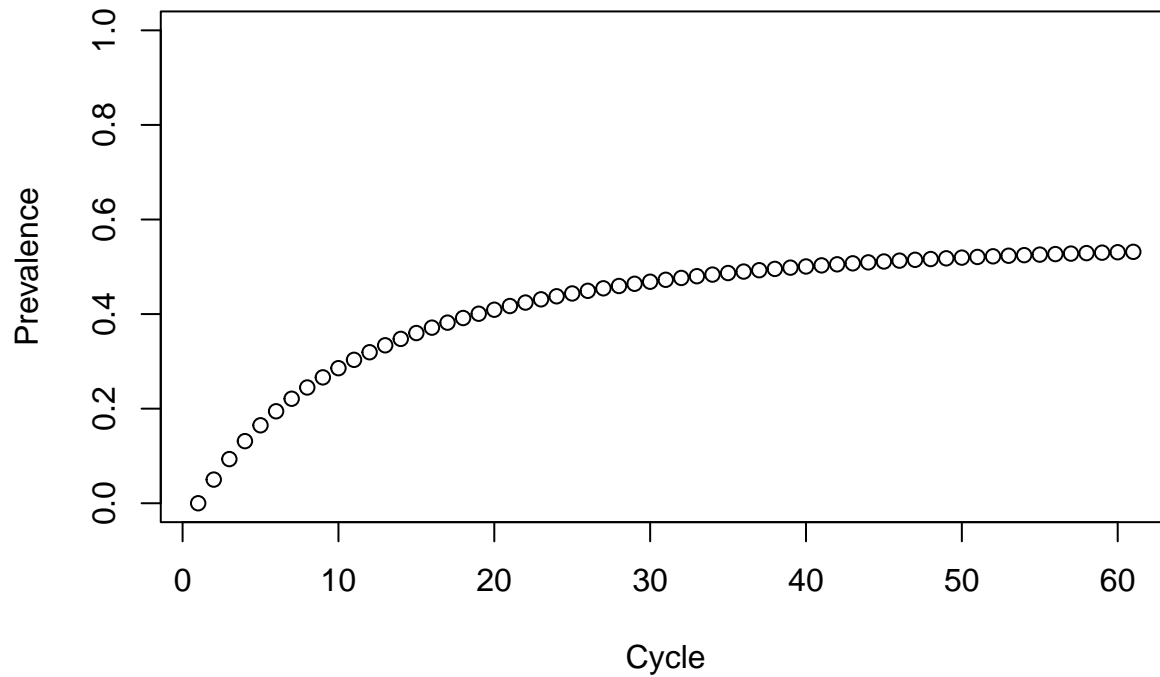
### 06.2.1 Life Expectancy (LE)

```
le_SoC <- sum(v_os_SoC) # summing probability of OS over time (i.e. life expectancy)
```

### 06.2.2 Disease prevalence

```
v_prev <- m_M_SoC[, "Sick"]/v_os_SoC
plot(v_prev,
     ylim = c(0, 1),
     ylab = "Prevalence",
     xlab = "Cycle",
     main = "Disease prevalence")
```

## Disease prevalence



## 07 Compute expected outcomes (costs, QALYs)

```
# Create empty vectors to store total costs and QALYs
v_tot_cost <- v_tot_qaly <- vector(mode = "numeric", length = n_str)
names(v_tot_qaly) <- names(v_tot_cost) <- v_names_str

### Expected discounted cost for each strategy
v_tot_cost["Standard of Care"] <- t(m_M_SoC %*% v_c_SoC) %*% (v_dwe * v_wcc)
v_tot_cost["Treatment A"] <- t(m_M_trtA %*% v_c_trtA) %*% (v_dwe * v_wcc)
v_tot_cost["Treatment B"] <- t(m_M_trtB %*% v_c_trtB) %*% (v_dwe * v_wcc)

### Expected discounted QALYs for each strategy
v_tot_qaly["Standard of Care"] <- t(m_M_SoC %*% v_q_SoC) %*% (v_dwe * v_wcc)
v_tot_qaly["Treatment A"] <- t(m_M_trtA %*% v_q_trtA) %*% (v_dwe * v_wcc)
v_tot_qaly["Treatment B"] <- t(m_M_trtB %*% v_q_trtB) %*% (v_dwe * v_wcc)
```

## 08 Cost-effectiveness analysis (CEA)

```
## Incremental cost-effectiveness ratios (ICERs)
df_cea <- calculate_icers(cost      = v_tot_cost,
                          effect    = v_tot_qaly,
                          strategies = v_names_str)

df_cea
```

```
##
## Strategy Cost Effect Inc_Cost Inc_Effect
## Standard of Care Standard of Care 8574.738 12.79462 NA NA
## Treatment B Treatment B 32745.314 17.16887 24170.58 4.374253
## Treatment A Treatment A 18272.100 13.94028 NA NA
##
## ICER Status
## Standard of Care NA ND
## Treatment B 5525.647 ND
## Treatment A NA ED
```

*## CEA table in proper format*

```
table_cea <- format_table_cea(df_cea)
table_cea
```

```
##
## Strategy Costs ($) QALYs Incremental Costs ($)
## Standard of Care Standard of Care 8,575 12.79 <NA>
## Treatment B Treatment B 32,745 17.17 24,171
## Treatment A Treatment A 18,272 13.94 <NA>
##
## Incremental QALYs ICER ($/QALY) Status
## Standard of Care NA <NA> ND
## Treatment B 4.37 5,526 ND
## Treatment A NA <NA> ED
```

*## CEA frontier*

```
plot(df_cea, label = "all", txtsize = 14) +
  expand_limits(x = max(table_cea$QALYs) + 0.1) +
  theme(legend.position = c(0.8, 0.3))
```

