PK8207 - Lecture memo

Jørn Vatn
Email: jorn.vatn@ntnu.no

Updated 2020-08-23

Markov State Model - An introduction

Introduction

Consider a stochastic process $\{Y(t), t \in \Theta\}$, where $Y(t)$ describes the state (deterioration level) of an item at time t. In the following we will assume that the state variable only takes a finite number of states. We first present the model when no maintenance is carried out, i.e., we start at time $t = 0$ and observe the system until failure. Let:

$$\begin{aligned} Y(0) &= y_0 \\ Y(T) &= y_r \end{aligned} \tag{1}$$

where T per definition is the time of the first failure. Between y_0 and y_r there are $r-1$ intermediate sates. By choosing a large value of r we could obtain a very good approximation to a continuous process if this is required. We will now let $\tilde{T}_i, i = 0,\ldots,r-1$ be sojourn times, i.e., how long the system stay in state i. Notationally we will typically denote the states by their number rather than by the value to simplify notation.

For the initial model we assume that the sojourn times are independent and exponentially distributed with parameter λ_i. Later on we will investigate how sojourn times may be modelled by arbitrary distributions. We also assume that the process runs through all states chronologically from y_0 to y_r without "stepping back" at any time.

Before we present the modelling framework for this simple situation we introduce the maintenance model. Figure 1 depicts the development of $Y(t)$ as a function of time. On the x-axis it is indicated that the system is inspected at period of times $\tau, 2\tau, 3\tau, \ldots$. If the system is found in state $Y(t) \geq y_l$ at an inspection, the system is renewed to an as good as new state, i.e., y_0.

We now go back to the simple situation where maintenance is not considered. Let $P_i(t)$ denote the probability that the system is in state i at time t. By standard Markov considerations we obtain the Markov differential equations:

$$P_i(t+\Delta t) \approx P_i(t)(1-\lambda_i \Delta t) + P_{i-1}(t)\lambda_{i-1}\Delta t \tag{2}$$

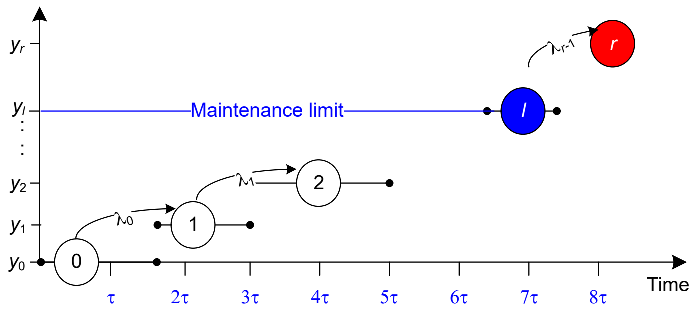

Figure 1: Markov transition diagram

where Δt is a small time interval and and we set $\lambda_{-1} = 0$ per definition. Further the initial conditions are given by:

$$\begin{aligned} P_0(0) &= 1 \\ P_i(0) &= 0 \text{ for } i > 0 \end{aligned} \tag{3}$$

Equation (2) could easily be integrated by a computer program, for example VBA in MS Excel. It is now easy to find MTTF by another integration, i.e.,

$$\text{MTTF} = \int_0^\infty [1 - P_r(t)]\, dt \tag{4}$$

and we should verify that we get $\text{MTTF} = \sum_{i=0}^{r-1} \lambda_i^{-1}$. Note that the transition rates, λ_i's, are assumed to be known, that is either they are estimated from data, or found by expert judgement exercises.

Exercise

Assume $r = 5$ and $\lambda_i = 0.01, i = 0, 1, \ldots$. Integrate the Markov differential equations and obtain the expected value and variance of the time to failure. Hint: Use partial integration for the variance similar to $\text{MTTF} = \int R(t)dt$. □

Equation (2) may be used in situations where we only allow transitions from state i to state $i+1$. In more general situations there could be transitions in principle from any state i to state j. In this situation we need to work with matrices. Let $\mathbf{A}$ be an $(r+1) \times (r+1)$ matrix where element (i, j) represents the constant transition rate from state i to state j. The indexing here starts at 0, e.g., $\mathbf{A}(0,1) = a_{0,1}$ is the transition from state 0 to state 1.

Further, let $\mathbf{P}(t)$ be the time dependent probability vector for the various states defined in $\mathbf{A}$. We now let $\mathbf{P}(t = 0) = [1, 0, 0, \ldots, 0]$ to reflect that the system starts in state 0. From standard Markov theory we now need the Markov differential equations, i.e., $\mathbf{P}(t) \cdot \mathbf{A} = \dot{\mathbf{P}}(t)$, from which it follows:

$$\mathbf{P}(t + \Delta t) \approx \mathbf{P}(t)[\mathbf{A}\Delta t + \mathbf{I}] \tag{5}$$

where Δt is a small time interval. Equation (5) is now used repeatedly to find the time dependent solution for the entire system. This corresponds to integrating Equation (2).

We now outline the main principle for working with matrices to find the time dependent solution and other relevant quantities. Assume we have access to a small library of matrix routines:

```
Function mMult(A,B) -> Returns a matrix equal to  A * B
Subroutine fixA(A) -> Fill diagonal of A such that sumrow=0
Function getIntMatrix(A, DeltaT) -> [A * DeltaT + I]
```

In the following we assume that the matrix library is defined by standard indexing, i.e., the first row is denoted row number 1 and so on. A warm up exercise to find MTTF is now:

```
Function getMTTF(A)
fixA A
MTTF = initial guess
DeltaT = MTTF / 1000
hlp = 0
t=0
P=[1,0,0,....]
IM = getIntMatrix(A, DeltaT)
Do While t < 5*MTTF
   P = mMult(P, IM)
   hlp = hlp +  (1-P(r+1)) * DeltaT
   t = t + DeltaT
Loop
getMTTF = hlp
End Function
```

To get higher precision we could increase the integration to e.g., 10MTTF. Note the motivation for this approach is given by:

$$\text{MTTF} = \int_0^\infty R(t)dt = \int_0^\infty [1-P_r(t)]\,dt \tag{6}$$

where $1-P_r(t)$ is the probability that we are not in state r at time t.

Exercise

Assume $r = 5$ and $\lambda_i = 0.01, i = 0,1,\ldots$ (time unit weeks). Find MTTF by numerical integration. Compare with the analytical result. □

So far the maintenance regime is not reflected in the approach. Let $\lambda_\mathrm{E}(\tau,l)$ be the effective failure rate, i.e., the expected number of failures per unit time if the system is inspected every τ time unit, and renewed whenever $Y(t) \geq y_l$ at an inspection. In the integration of Equation (5) we start with $t = 0$ and whenever t coincides with τ, 2τ etc., special actions are taken:

```
Function lambdaEffective(A,tau,l)
fixA A
MTTF = getMTTF(A)
DeltaT = MTTF / 1000
hlpF = 0
t=0
localTime=0
P=[1,0,0,....]
IM = getIntMatrix(A, DeltaT)
Do While t < 10*MTTF
   P = mMult(P, IM)
   hlpF = hlpF + P(r + 1)      Add to effective failure rate
   P(1) = P(1) + P(r + 1)      If system is failed, it is assumed to be renewed
   P(r + 1) = 0                Clear probability
   If localTime >= tau Then
     sumP = 0
     For i = l+1 To r
         SumP = SumP + P(i)
         P(i)=0
      Next i
      P(1) = P(1) + SumP
      localTime = 0
   Else
      localTime = localTime + DeltaT
   End If
   t = t + DeltaT
Loop
lambdaEffective = hlpF / t
End Function
```

Note the indexing, i.e., the failed state is $r+1$ and the maintenance limit is $l+1$.

In the `If localTime = tau` part of the script above we have used a loop to simulate what is happening during an inspection. A more efficient way to do this would be to create an "inspection matrix", say $\mathbf{M}$ defined by:

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \\ 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ \vdots & & & & \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{7}$$

where the starting point is an identity matrix, but where we from the row corresponding to state l shift the "ones" to the left.

```
:
If localTime >= tau Then
   P = mMult(P, M)
   localTime = 0
Else
:
```

Such an inspection matrix could also be used to specify that an inspection is not perfect. For example if q is the probability that an inspection fails to reveal that the actual state is l or higher, the corresponding leftmost "one" is replaced by $1-q$ and the diagonal element is replaced by q for rows corresponding to states $l, l+1, \ldots, r-1$. A inspection matrix could also be used to specify that upon an inspection it might be decided to repair to a state which is not as good as new. For example in 80% of the cases we repair to state 0, in 15% of the cases we repair to sate 1 and in 5% of the cases we repair to state 2.

Exercise

Assume $r = 5$ and $\lambda_i = 0.01, i = 0, 1, \ldots$ (time unit weeks). Assume the system is inspected with intervals of length $\tau = 26$. If the system is found in state $l = 4$ the system will be renewed. Renewal takes place immediately. The probability that a inspection reveals that the system is in state $l = 4$ is 70% when this is the case. Find the effective failure rate for this situation. □

Significant repair times

So far we have assumed that repair times could be neglected. If we can not neglect repair times we need to model repair times in the transition matrix **A**. For example if at an inspection we with some probability q will decide to repair from state i to state j with constant repair rate μ a first approach would be to modify the **A**-matrix, i.e., $\mathbf{A}(i,j) = a_{i,j} = q\mu$. However, this would imply that a repair starts immediately after the system has reached state j. In reality, a repair can first start after the coming inspection.

To handle the situation we now introduce "virtual" states. A virtual state is a state in the **A**-matrix representing the situation where a maintenance action has been decided and the repair is actually started. For each pair (i,j) where there could be a repair from state i to state j a virtual state $k_{i,j}$ is defined. Then the associated transition rate is set to $a_{k_{i,j},j} = \mu$. The row and column representing the virtual state $k_{i,j}$ can be any ones larger than those already "occupied". The inspection matrix **M** will also get an additional row

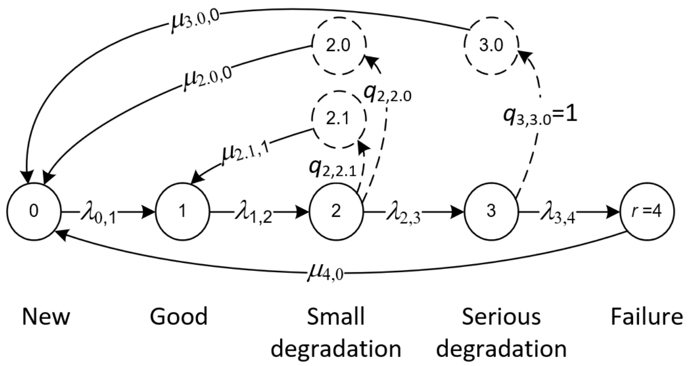

Figure 2: Markov transition diagram with potential repairs

and column representing the virtual state $k_{i,j}$, where $\mathbf{M}(i,k_{i,j}) = q$, where we in addition need to ensure that the row sum equals one.

Note that while repairing from state i to state j represented by $a_{k_{i,j},j} = \mu$ there might be a "competing" transition from for example state i to state l, thus we also need to specify $a_{k_{i,j},l} = \lambda_{i,l}$.

Figure 2 illustrates the full Markov diagram for a situation with $r = 4$. Here λ_{i_j} is the transition rate from state i to state j representing degradation. Further μ_{i_j} is the repair rate from state i to state j. When a repair is initiated as a result of a proof-test, virtual states are introduced. For example state (2,1) represent that it after a test is decided to repair from state 2 to state 1. The doted lines represent transitions that instantaneously take place after a proof-test. The probabilities given by the q-values represent maintenance decisions. For example $q_{3,3.0} = 1$ represents that if a state 3 is revealed by a proof-test, we always initiate a repair to state 0. $q_{2,2.0}$ is representing the probability that we after revealing a state 2 on a proof-test we initiate a repair to state 0. The q-values are entered into the inspection matrix, $\mathbf{M}$. Note that in Figure 2 we use the notation $a_{\text{From,To}}$ without indicating the actual row and column numbers in the transition matrix. The notation $a_{k_{i,j},l}$ on the other hand, is used to identify a row and column number in a matrix in the code.

In previous sections we have focused on the effective failure rate, but we might also be interested in the average portion of time we are in each state. For example we may use:

```
:
Do While t < 10*MTTF
   P = mMult(P, IM)
   Pavg = Pavg + P
   If localTime >= tau Then
      P = mMult(P, M)
      localTime = 0
```

```
   Else
      localTime = localTime + DeltaT
   End If
   t = t + DeltaT
Loop
Pavg = Pavg * DeltaT / t
:
```

Exercise

Assume $r = 5$ and $\lambda_i = 0.01, i = 0,1,\dots$ (time unit weeks). Assume the system is inspected with intervals of length $\tau = 26$. If the system is found in state $l = 4$ the system will be renewed. There is a *logistic delay* of in average 4 weeks before the repair takes place. Delay time is assumed to be exponentially distributed. The probability of revealing state $l = 4$ is still 70%. Find the effective failure rate for this situation. $\square$