A Comprehensive Review and Simulation of George Popov's Markov Processes through Chains with Complex States.

STAT 403: Stochastic Processes

Alejandro Builes Gallego (48801039)

The University of British Columbia

May 11, 2024

Summary of the Topic

Introduction

This paper explores a simulation method for Markov chains, specifically focusing on complex systems' reliability simulations. Traditional methods require transforming and mathematically computing state transitions within Markov chains. Instead, this approach merges partial states into combined entities known as complex states, eliminating the need of chain conversions and mathematical calculations.

George Popov has used the General Purpose Simulation System (GPSS), a simulation language designed for discrete-event simulations, to improve the simulation process of Markov chains. This method makes it easier to create, modify, and run Markov chains using the GPSS platform, which is ideal for handling complex systems and processes. The effectiveness of this approach has been confirmed in his paper through comparisons with both analytical and traditional simulation methods, demonstrating its reliability in producing accurate results.

Author's Analytical Proposed Model

In this paper, the author is looking at a system that relies on two separate power sources to stay operational, similar to having two different battery packs. This system can exist in one of four states:

- **State 1 (AA):** Both power supplies are working.
- **State 2 (FA):** The first PSU is working, but the second has failed. The system is still on because it has one working PSU.
- **State 3 (AF):** The first PSU has failed, but the second is working. The system is still on because it has one working PSU.
- **State 4 (FF):** Both power supplies have failed.

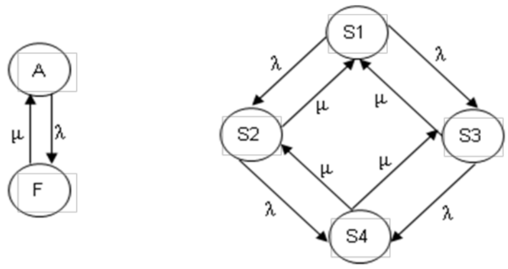

Figure 1: Graph state of one power supply and the entire reserved system

The Markov chain is defined by the transition rates λ (failure rate) and μ (recovery rate), with transitions being time-independent, a characteristic of homogeneous processes.
For this system, the transition frequency (H_{ij}), which is the likelihood of moving from one state to another in a unit of time, is expressed as:

$$H_{ij} = \frac{1}{T_{ij}} = \lambda_{ij} \cdot P_i$$

where P_i is the probability of the system being in state S_i, λ_{ij} is the transition rate from state S_i to state S_j, and T_{ij} is the expected time to transition from S_i to S_j.

The model simplifies into a reduced graph (shown in Figure 2), streamlining the calculation of system reliability. It translates the original four-state diagram into a more manageable form, reflecting the system's state as either having both PSUs available (AA), one PSU available (AF or FA), or both failed (FF).

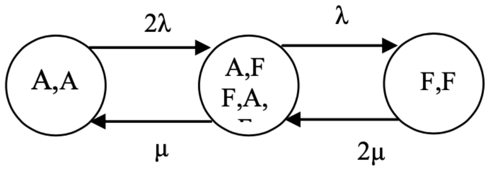

Figure 2: Reduced graph of reserved power

Using this reduced graph, the probability that the system remains operational (in states AA, AF, or FA) is given by:

$$P_{AA,AF,FA} = \frac{\mu_1\mu_2 + \mu_1\lambda_2 + \lambda_1\mu_2}{(\mu_1 + \lambda_1)(\mu_2 + \lambda_2)}$$

This equation provides the overall probability of the system being able to function with at least one PSU operational. In the specific case where both PSUs have identical reliability characteristics, the probability simplifies further to:

$$P_{AA,AF,FA} = \frac{\mu^2 + 2\mu\lambda}{(\mu + \lambda)^2}$$

Furthermore, the likelihood of a single PSU failure, critical for understanding system redundancy, is determined by:

$$P_{AF,FA} = \frac{2\mu\lambda}{(\mu + \lambda)^2}$$

Equally important is the probability of complete system failure, which is the worst-case scenario:

$$P_{FF} = \frac{\lambda^2}{(\mu + \lambda)^2}$$

Lastly, the system availability, representing both PSUs working correctly, is crucial for assessing the dependability of the power supply system:

$$P_{AA} = \frac{\mu^2}{(\mu + \lambda)^2}$$

Finally, from what is said in the paper high system operational availability is achieved when λ is much smaller than μ, meaning failures are infrequent and easily resolved. In contrast, when λ approaches μ, the risk of concurrent failures increases, threatening the system's stability.

Complex States

The model innovatively breaks down into complex states, simplifying the representation of a dual power supply system's various operational and failure scenarios.

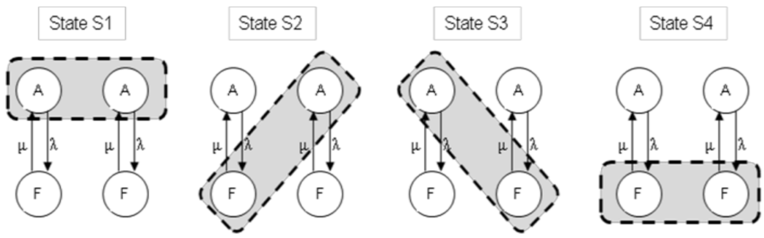

Figure 3: Complex representation of the state of the reserved system.

Figure 3 illustrates the complex state system, showing the same four possible conditions of the power supplies within the system. This complex representation shows the entire state space of the system and the potential transitions from failure and recovery rates (λ and μ).

This finishes up the summary of the topic. This next section will demonstrate how the theoretical concepts from Popov's paper can be observed through R simulations.

Problem Importance

The topic presented in the paper has significant importance for simulating and understanding system behaviors. The method Popov proposes helps quickly identify potential issues and improve system designs for operational efficiency and safety, as seen in power supply systems, for example. Using Markov chains to simulate complex systems enables people to understand where problems might occur, plan maintenance more effectively, and improve system designs to reduce risks. Additionally, modeling how states change over time and using techniques like Maximum Likelihood Estimation (MLE) makes predictions more accurate. This is especially valuable in industries where system failures can lead to substantial losses or safety issues.

Simulation Methodology

Data Collection

Data collection involves tracking the operational and failure states of a dual power supply system. For this part, the observed counts will be simulated in R to collect information and see how the system works.

The simulated transition counts are as follows:

	AA	AF	FA	FF
AA	339	97	60	0
AF	156	783	286	146
FA	0	491	669	496
FF	0	0	641	5835

These counts show the frequency of transitions between states during the simulation. High numbers on the diagonal indicate states where the system tends to remain, suggesting stability or difficulty in transitioning away from those states.

Analysis of Collected Data: Estimated Transition Matrix from MLE

The MLE-based estimated transition probabilities are shown in the following transition matrix and also in Figure 4:

	AA	AF	FA	FF
AA	0.6835	0.1956	0.1210	0.0000
AF	0.1138	0.5711	0.2086	0.1065
FA	0.0000	0.2965	0.4040	0.2995
FF	0.0000	0.0000	0.0990	0.9010

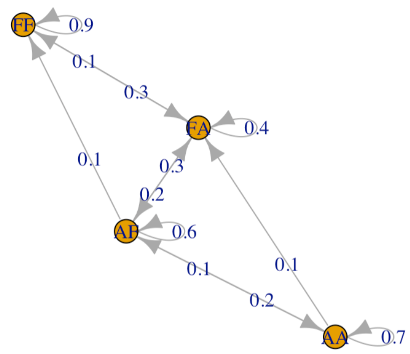

Figure 4: Diagram of Simulated State Transitions in the Markov Chain Model.

Interpretation and Summary of Results

- **State AA (Row 1):** There is approximately a 68.3% chance of remaining in state AA, with notable probabilities of transitioning to state AF (19.6%) and FA (12.1%).
- **State AF (Row 2):** State AF has a 57.1% chance of maintaining itself, with significant movements to FA (20.9%) and a non-trivial probability (10.6%) of deteriorating to state FF.
- **State FA (Row 3):** State FA tends to remain in the same state about 40.4% of the time but shows nearly equal probabilities of transitioning to either AF or FF (approximately 30% each).
- **State FF (Row 4):** There is a very high probability (90.1%) of staying in state FF, with a small likelihood (9.9%) of moving back to FA.

The analysis of the simulated transition counts and the estimated transition matrix reveals key insights. The results indicate the following:

- State **AA** suggests that while the state is somewhat stable, it is quite likely to transition to other states.
- State **AF** is pivotal with decent stability but a considerable risk of worsening, suggesting a need for preventive measures.
- State **FA** represents a critical moment, displaying significant transitions to both less severe and more severe states, emphasizing the need for monitoring and quick interventions.
- State **FF** appears as an absorbing or terminal state indicative of system failure or critical condition that is challenging to recover from, emphasizing strategies to prevent entering this state.

Extension of Proposed Methodology

Testing the Markov chain model with real-world data is key to making it useful. Initially, we might use simulated data to build the model, but real data from actual systems is crucial for refining it. This helps us ensure the model's predictions are accurate. By using real data, we can fine-tune the model and make it reliable for important decisions, like predicting system failures or planning maintenance. For further investigations, I would like to work with real data to observe how power systems behave and how Markov chains can explain that behavior.

References

- Popov, G. (2018, June). Simulation of Markov Processes through Chains with Complex States. In Proceedings of the 2018 HiTech Conference. IEEE. https://doi.org/10.1109/HiTech.2018.8566393

Code/Data Appendix

```r
library(markovchain)

transitionMatrix <- matrix(c(
  0.7, 0.2, 0.1, 0.0,  # Probabilities from State AA to others
  0.1, 0.6, 0.2, 0.1,  # Probabilities from State AF to others
  0.0, 0.3, 0.4, 0.3,  # Probabilities from State FA to others
  0.0, 0.0, 0.1, 0.9   # Probabilities from State FF to others
), byrow = TRUE, nrow = 4, dimnames = list(c("AA", "AF", "FA", "FF"), c("AA", "AF", "FA", "FF")))

mc_sim <- new("markovchain", states = c("AA", "AF", "FA", "FF"), byrow = TRUE, transitionMatrix = initial_transition_matrix)

set.seed(403-560)

# Simulate the Markov chain for 10000 steps starting from 'AA'
simulated_states <- rmarkovchain(n = 10000, object = mc_sim, t0 = "AA")

# Create the observed transition count matrix from the simulated states
observed_counts <- matrix(0, nrow = 4, ncol = 4, dimnames = list(c("AA", "AF", "FA", "FF"), c("AA", "AF", "FA", "FF")))
for (i in seq_along(simulated_states) - 1) {
  from_state <- simulated_states[i]
  to_state <- simulated_states[i + 1]
  observed_counts[from_state, to_state] <- observed_counts[from_state, to_state] + 1
}

# Define the likelihood function for MLE
likelihood_function <- function(params) {
  # Ensure parameters are in the range [0, 1] and sum to 1 for each row
  matrix_probs <- matrix(params, nrow = 4, byrow = TRUE)
  row_sums <- rowSums(matrix_probs)
  matrix_probs <- sweep(matrix_probs, 1, row_sums, FUN = "/")

  # Avoid log of zero by ensuring all probabilities are at least very small
  matrix_probs <- pmax(matrix_probs, .0001)

  # Calculate the log-likelihood of observed transition counts
  log_likelihood <- sum(observed_counts * log(matrix_probs))
  return(-log_likelihood)  # Minimize the negative log-likelihood
}

# Initial guesses for optimization (uniform probabilities)
initial_values <- rep(1/4, 16)

# Optimizing the likelihood function to find MLE of transition probabilities
optim_results <- optim(par = initial_values, fn = likelihood_function, method = "L-BFGS-B", lower = rep(0, 16), upper = rep(1, 16))

optimized_params <- optim_results$par
estimated_transition_matrix <- matrix(optimized_params, nrow = 4, byrow = TRUE)
normalized_matrix <- sweep(estimated_transition_matrix, 1, rowSums(estimated_transition_matrix), FUN = "/")

```

```
cat("Simulated Transition Counts:\n")
print(observed_counts)
##       AA  AF  FA   FF
## AA 339  97  60    0
## AF 156 783 286  146
## FA   0 491 669  496
## FF   0   0 641 5835
cat("\nEstimated Transition Matrix from MLE:\n")
print(normalized_matrix)
plot(mc_sim)
```