Dual Policy Inventory System

IMSE-866 Stochastic Processes

Blake Conrad

11/20/2018

# Executive Summary

With extensive research done on the utility of the dual policy inventory system, we discovered appropriate conditions with which this can be modeled using mathematical modeling. This finding obtains the feasible input into a manageable space that enables a data driven approach to solving the problem. Final results show that with the assumptions, parameters, and the model itself, we see an expected cost in the system of 11.6. In essence, we can expect to see a average reorder cost of about 11.6, which is greater than both the cost of subcontractors and manufacturers, displaying a dilemma with our current policy. The key problem in our system is the backorders; subcontractors not being able to keep up with the demands caused the backorders. The primary cause of this is the rate at which subcontractors can resupply to meet demand. Several solutions exist to mitigate this problem: Consult faster subcontractors, invest more resources into the manufacturers to make them equivalent or better than the subcontractors, mutate our dual policy architecture approach to one that is more appropriate to our problems we are facing, which could include a single subcontractor working for both parts. Any of these solutions could save time and money when dealing with multiple customers and a higher demand than supply problem we are facing. This analysis could be augmented with predictive analysis to determine what best next steps could be for our procurement processes given historical information or an in house simulation to explore an environment of decisions to yield an optimal reward for the company.

# Problem Description

In order to cut costs and reduce lead times, many manufacturers design their products and processes so that the final product can be quickly assembled from its components. These systems are commonly referred to as assemble-to-order (ATO) systems. ATO systems combine the benefits of make-to-order MTO systems and make-to-stock (MTS) systems to provide custom products at short lead times. The strategy initially found popularity in the computer industry and since then the concept has gained acceptance in several other industries. An ATO system assembles a single product with two components. Component k=1,2 can be manufactured by the in house manufacturing facility Mk and the local subcontractor Sk. The components are stored at inventory location Lk and are assembled at station A to satisfy the demand for the final product.

Let, representing each of our components.

Let , representing time.

Distribution of subcontractors and manufacturers that can resupply inventory:

Where,

Costs associated with each of our re-suppliers:

Where,

Distribution of demand over time:

Inventory refreshment point for both components:

Inventory resupply point for both components:

* , Value when inventory drops below we must reorder

General functions:

* , Inventory for component k at any given point in time.
* , Number of orders for component k at any given point in time.
* , Number of backorders for component k at any given point in time.

Balancing equations:

* , Supply = demand equations.

Constraints:

* If we must resupply. We may resupply with either or
* If we can resupply. We can only resupply with

Constructed Variables:

* , Distance to our resupply point
* , Inventory amount we must resupply if we are at

Stochastic Process:

* , a 6-tuple representing the status of the system at any point in time.
  + Is the total number of units of inventory needed to completely replenish to
  + Is the total amount of inventory in the system
  + Is the total amount of backorders in the system

By defining our system in such a way, we know that the sum of each of our six indices should sum to zero across every state, at any point in time, in the CTMC. We now begin with the necessary assumptions for our analysis.

Since looking at the problem in terms of inventory replenishments and demands over time, we will find that we are a time dependent CTMC. Rather, by looking at the problem in terms of residuals, we will be able to find patterns, and hence construct our CTMC in a predicable manner. In order to do so, we must look back at our previous equations listed above:

* ,

With these, we are able to look at the distance, or number of transitions, that are required for before we need to replenish.

1. We must assume that .
2. We must also assume that .
3. We must assume that when a customer demands a unit from *A*, it requires exactly one component from and exactly one component from .
4. We must assume that resupply happens at the same time, and is instantaneous for both components.
5. We assume that backordering begins when our inventory dips below the resupply point; in terms of cost, this is essentially the assumption that is needed to penalize any orders placed below our thresholds.

These assumptions ensure consistency and manifest patterns that enable discovery of a static CTMC structure and transition probability matrix along with steady states.

# Model Analysis

We construct the transition probability matrix using layers. Assume our store has and in inventory, 0 orders sent for , alongside no backorders initially. Hence our initial state is . The notation used will use for our initial layer of simple demands until, we transition to a new layer with two options; full part 1 replenishment or backorder (because we assumed . These are in the next alpha character respectively. We allow to progress transitions (because , we guarantee we hit before we loop again) and to progress an infinite number. Next we progress to our final layer that enables a pattern. will transition off to another pair of states that enable replenishment or backorder. From we progress to the next alpha character . We allow to progress transitions and to progress an infinite number. We now notice that if we continue our pattern becomes an identical state to , so we construct the simple transition from causing our structure in the CTMC. Hence, we let Q be represented as a matrix in *Table*

*Table 1:* Transition Probability Matrix for, *where \* represents the sum of the current row and empty cells are zero.*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A0 | A1 | A2 | … |  | B0 | B1 | … |  |  |  | … |  | C0 | C1 | … |  |  |  | … |  |
| A0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A1 |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A2 |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| … |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| B0 |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| B1 |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| … |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |
| … |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |
| C0 |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |
| C1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |
| … |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |
| … |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |

Our goal now is to solve the following equation:

Where the vector p must satisfy the following:

We now list our initial equation layout:

1. ,

Now we lay out each within our steady state constraint:

Notice we have five key segments to our equations, so we individually assess their decompositions in the steady state constraint:

After extensive algebra it can be shown that as k approaches infinity, making this analysis possible.

After solving our stead state equations we obtain the following general solution:

1. ,

By knowing, we can show that algebraically, every other term in our steady state equations is dependent on it and only it. Now that we have our steady state equations, with some provided parameters, we can answer questions about cost, inventory levels, and much more.

# Results

Provided the following information, we now continue to construct performance attributes associated with our model:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Therefore, we begin by finding what our steady state vector looks like, followed by finding the expected cost for our system.

Assume a maximum of only three backorders can take place. Our steady state vector becomes the following after plugging in all of our terms:

We know the costs associated with any state, so we simply take the dot product between our cost vector and our steady state vector:

We used the *Python* programming language to calculate the results.

# Conclusion

With extensive research done on the utility of the dual policy inventory system, we discovered appropriate conditions with which this can be modeled using Continuous Time Markov Chains. The findings here obtain the feasible input into a manageable space that enables a data driven approach to solving the problem. Final results show that with the appropriate model and provided parameters, expected cost in units were at 11.6. This means we would expect to see a cost of about 11.6 among our transitions. Since , we can see that we are paying for backorders, driving up our overall cost. This can be mitigated in several ways, one of which is to get faster subcontractors. With faster subcontractors, we will no longer have the problem of backorders under our current model assumptions, because the birth rate (demand) will always be slower than our death rate (resupply).