A Stochastic Optimal Control Model for Supply Networks with Correlated Demand and Transshipments

Project at the Institute for Operations Research
ETH Zurich

Julien Uffer, Faris Cassim and Jacques Chastenet
Supervisor: Dr. Marco Laumanns

April 15, 2008
Abstract

The aim of this project is to understand the different characteristics of supply chains with unknown demand. We will study the presence of correlated demand with respect to time as well as location, the influence of backlogs (e.g., the possibility that the demand stays partially unfilled), and finally the benefits of the possibility for two retailers with a common supplier to exchange goods. Our main goal is to create and understand a model of a supply chain with these characteristics and to investigate the value of the transshipment option in different settings.

Acknowledgments

We thank Dr. Marco Laumanns for his invaluable help.
Introduction

A supply chain, logistics network, or supply network is the system of organizations, people, activities, information and resources involved in moving a product or service from supplier to customer.

Supply chains are of great importance for manufacturing companies today. An efficient supply chain can make the difference between glory and failure. That is why successfully modeling supply chains and optimizing their efficiency has become essential.

Friedman [4] explains that successful supply chaining has brought millions even billions of dollars to companies such as Walmart. Walmart is the biggest retail company of the world and it does not make a single thing. All it makes is a hyper-efficient supply chain! Companies both can and must take advantage of the best producers at the lowest prices anywhere they can be found. If they don’t, their competitors will. It looks like an easy process, but making these supply chains work is much harder than it looks and requires constant innovation and constant adjustment.

Research and development in this area is therefore essential. Quantitative skills have become ever so important in modeling and implementing supply chains. Many areas in operational research are indeed dedicated to optimizing supply chains. This project focuses on one specific area: optimal control policies. We consider a retail company that faces constant or fluctuating demand and try to figure out an optimal ordering policy (from an internal or external supplier), in other words an ordering policy that will minimize cost therefore maximizing profits.

The basis of this project is a particular optimal control approach for supply networks in [1,2]. We will extend the models presented in these two papers (that are presented in Chapter 1) by taking into account the following aspects: Correlated demand, the option of backlogs, and the option of transshipments.

Correlations in demand can have two different forms: time correlation and locational correlation. The former means that the demand tomorrow depends on the demand today and the latter means that the demand in one retailer is correlated with the demand of another retailer in the same region.

Backlogs are also an important extension of the model. In the models presented in the papers above, demand had to be satisfied at each time step. By introducing backlogs, the retailer can have the option of not satisfying demand at the expense of losing the customer or paying a penalty.

The examples studied in the above papers consist of one retailer with one supplier.
Introducing transshipments to our model means permitting two retailers to exchange goods between themselves. This could lead to other results and other policies. We will indeed verify if this is true. We will also take interest in evaluating the cost of this transshipment option.

We will first present these model extensions and their consequences separately before attempting to create and analyze a model where different upgrades are taken into account at the same time.
Chapter 1

An optimal control model for supply networks

1.1 The setup

In this section we will introduce the dynamics of the supply chain model and all what the reader should know in order to understand the following parts. Consider a supply chain represented by a graph with nodes denoting different actors (customers, retailers, wholesaler) connected by edges which can take different time values. First of all, in order to derive a first order difference equation model for the dynamics of the system we need to add some nodes to our network model such that the transportation time over each arc is exactly one unit. In all our graphs we will draw those additional nodes as circles, see Fig. 1.1, so that one can really differentiate the real actors of the supply chain from the additional nodes when looking at the graphs.

\[ \begin{align*}
D & \quad X_1 \quad 1 \quad W \\
D & \quad X_1 \quad X_2 \quad W
\end{align*} \]

Figure 1.1: The graph on the top shows the initial supply chain, the one on the bottom shows the modified supply network with every edge’s value equal to one. Here \( W \) represents a supplier, \( X_1 \) a shop and \( D \) the end customer that generates the demand.

So the supply network is represented as a connected graph \( G = (V, S) \) where \( V \) is the set of all the vertices of the graph (including additional nodes) and where \( S \) represents the set of all its edges.
1.2. THE SYSTEM’S DYNAMICS

Definition 1. The backward arcs in the supply chain are defined as the information arcs. The information arcs can be divided into two distinct groups: the demand arcs, which are the backward arcs that come out of the customers, and the control inputs, which are the rest of those edges. We will denote the set of all the information arcs by \( I \), they will be represented by dashed arrows.

Definition 2. The forward arcs are the material arcs, i.e., the arcs in which the material flows between actors of the supply chain. The set of those edges will be denoted by \( M \), they will be represented by solid lines.

It is important to bear in mind that all the orders are made through the information arcs (demand and control input). Once the order is made through those arcs, the ordered goods will flow (not necessarily instantaneously) into the corresponding transportation path.

Definition 3. The state of the system at time \( t \) is a vector \( x(t) \in \mathbb{R}^n \) where \( x_i(t) \) is the inventory of node \( i \) at time \( t \) and where \( n \) is the number of nodes in the network minus the number of consumers and the number of suppliers.

Our final goal being to present optimal policies, it is important for the reader to know the dynamics of the system. We argue, with respect to what we introduced above, that the state at time \( t + 1 \), \( x(t + 1) \), is a function of the state at time \( t \), \( x(t) \), of the control input at time \( t \), \( u(t) \), and the disturbance at time \( t \), \( d(t) \). We can thus write the following:

\[
x(t + 1) = f(x(t), u(t), d(t)),
\]

where \( x(t) \in \mathbb{R}^n \), \( u(t) \in \mathbb{R}^{n_u} \) and \( d(t) \in \mathbb{R}^{n_d} \). The number \( n_d \) is the number of demands as defined above and \( n_u \) is number of control inputs, i.e. the number of backwards arcs minus the number of demands.

Remark 1. Here we suppose that the only disturbance of the system is the uncertain demand coming from the customers (which are defined to be the nodes without any outgoing material arcs).

1.2 The system’s dynamics

Now that we have found the overall system’s dependencies, we will concentrate on the dynamics of every single node of the network in order to derive a relationship between the state of the total system at time \( t + 1 \) and the state of it at time \( t \).

If we take an arbitrary node \( v \) of the supply chain and define its internal state at time \( t \) \( x_v(t) \), to be, according to the previous section, its inventory of goods at this time, then it is intuitive that its internal state at the next time step will be its initial inventory minus what this node gives to its successors through outgoing material arcs plus what it will receive from various predecessors through incoming materials arcs. In order to write this mathematically we should introduce the following:

- \( z_{in}^{(v)} \): flow on the incoming information arcs.
1.2. THE SYSTEM’S DYNAMICS

- $z_{out}^{(v)}$: flow on the outgoing information arcs.
- $y_{in}^{(v)}$: flow on the incoming material arcs.
- $y_{out}^{(v)}$: flow on the outgoing material arcs.

We can now write the dynamics for a single node $v$ of the network as

$$x^{(v)}(t + 1) = x^{(v)}(t) + 1 \cdot y_{in}^{(v)}(t) - 1 \cdot y_{out}^{(v)}(t).$$  \hspace{1cm} (1.2)

At this stage we can already simplify the dynamics of the additional nodes. As an additional node $w$ is only "transient", and by this word we mean that all its inventory will be shipped at the next time step, its new state (or inventory) is only what this node gets from its successors through incoming material arcs. This can be written as the following:

$$x^{(w)}(t + 1) = 1 \cdot y_{in}^{(w)}(t).$$

Note that the dynamics is similar for an additional node $w'$ in a path build up of information arcs is

$$x^{(w')}(t + 1) = 1 \cdot z_{in}^{(w')}(t).$$

The nodes $v_c \in V$ that have no outgoing material arcs are referred as the customers. They generate a demand $d^{(v_c)}$ through their outgoing information arcs to some retailers. Hence their dynamics is:

$$z_{out}^{(v_c)}(t) = d^{(v_c)}(t).$$

Some nodes $v_m \in V$ have no incoming material flows, they are a source of infinite supply capacity (we can think of them as factories that give us everything we order). Those nodes transform incoming orders into goods that will flow to their successors through material arcs. The dynamics of each of these nodes is then given by

$$z_{in}^{(v_m)}(t) = y_{out}^{(v_m)}(t).$$

Remark 2. Sources and sink nodes (customers) have no internal state variables, that is why we did not include them in the dimension of the system state $x(t)$ in (1.1).

We have found the dynamics of every node of the supply network and it is time to give the dynamics of the whole system. If we denote, as above, the control input by $u(t)$ and the disturbance vector, which in our case is the demand vector, by $d(t)$ then the state of the system at time $t + 1$ is given by

$$x(t + 1) = Ax(t) + Bu(t) + Ed(t),$$  \hspace{1cm} (1.3)

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times n_u}$ and $E \in \mathbb{R}^{n \times n_d}$. 

7
1.3. DYNAMIC PROGRAMMING

Example 1. We now give a short example so that the reader can get familiar with what we introduced above. Consider a simple supply chain with one consumer, one retailer and one supplier. The retailer receives its supply from the supplier $W$ with a shipping delay of one unit of time and the ordering delay is also one unit. Hence according to what we said above we should introduce two additional nodes as shown in Fig 1.2.

![Figure 1.2: simple supply chain.](image)

Here by computing the dynamics of each node we can find the matrices of (1.3):

$$
A = \begin{pmatrix}
1 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & 0
\end{pmatrix},
B = \begin{pmatrix}
0 \\
0 \\
1
\end{pmatrix},
E = \begin{pmatrix}
-1 \\
0 \\
0
\end{pmatrix}.
$$

1.3 Dynamic Programming

We shall now introduce some facts about what we are willing to find by modeling the dynamics of the system. Think of a company owning all the supply chains of one of its products and planning its production for a certain discrete time period. The goal is clearly to minimize the cost under certain conditions such as always fulfilling demand. Of course a new decision must be made on each time step as the demand is unknown. But, if we know its probability distribution, using stochastic dynamic programming, we can compute, on each time step, the worst case cost and the expected case cost and give an optimal ordering policy for this stage.

To introduce stochastic dynamic programming we will consider the problem of managing a stock with unknown demand and a horizon of $N$ time steps. At the beginning of each time period we should decide which quantity to produce. Consider the following:

- Unlimited capacity of production.
- Capacity of stockage: $S$.
- No cost of production.
- $h_i(y)$ is the holding cost of $y$ articles in the period $i$.
- The demand at period $i$ is a discrete random variable $D_i$ and has to be satisfied in each time period $i$. 

1.3. DYNAMIC PROGRAMMING

- \( D_i^{\text{max}} \leq S + D_i^{\text{min}}. \)

Furthermore we assume that the probability that \( D_i = d \) is given and that we do not allow any delivery delays. The goal is to minimize the expected overall cost on a \( N \) time period.

Let’s write \( u(i) \) the production at the beginning of period \( i \), and \( J_i(x(i)) \) the minimum of expected cost over the periods \( i, i-1, \ldots, 1 \) if the stock is \( x(i) \) at the beginning of period \( i \). The recursion relation is then:

\[
J_i(x(i)) = \min_{u(i)} \left\{ \sum_{D_i^{\text{min}} \leq d \leq D_i^{\text{max}}} \left[ h_i(x(i) + u(i) - d) + J_{i-1}(x(i) + u(i) - d) \right] \cdot P(D_i = d) \right\}
\]

Under the constraints that \( x(i) + u(i) - D_i^{\text{max}} \geq 0 \) and \( x(i) + u(i) - D_i^{\text{max}} \leq S. \)

In practice, if the model stays pretty simple (one consumer, one retailer, one supplier, ...), the equation above can be easily computed by hand. However the models we are willing to study are not so trivial and the computations become long and difficult. Therefore we will need the help of the computer.

More generally, the problem is to find an optimal control input \( u(t) \) to the system (1.3) with state cost vector \( q \in \mathbb{R}^n \) and ordering cost vector \( r \in \mathbb{R}^n \), such that those cost are given by \( q^T x(t) \) and \( r^T u(t) \), and with constraints:

\[
F x(t) + G u(t) \leq g, \quad F \in \mathbb{R}^{n_g \times n}, G \in \mathbb{R}^{n_g \times n_u}, g \in \mathbb{R}^{n_g}, n_g \in \mathbb{N}_0. \quad (1.4)
\]

**Remark 3.** The constraints could be used to impose upper and lower bounds on the state variables and the control variables or even to make the orders dependent on state.

Once we have those inputs and if we are given the initial state, the goal is to find an optimal policy for the \( N \) periods model, that is, to find an optimal sequence of control input \( (u(k))_{k=1}^N \) such that:

\[
J_k(x(k)) = \min_{u(k) \in U_k} \left\{ q^T x(k) + r^T u(k) + \mathbb{E}\{J_{k-1}(A x(k) + B u(k) + d(k))\} \right\} \quad (1.5)
\]

\[
\text{s.t.} \left\{ \begin{array}{l}
F x(k) + G u(k) \leq g \\
A x(k) + B u(k) \in X_{k-1}
\end{array} \right. \quad (1.6)
\]

where:

- \( U_k \) is the set of all possible order policies,

- \( X_k = \{ x \in \mathbb{R}^n : \forall d \in D \exists u \in \mathbb{R}^{n_u} \text{ with } F x + G u \leq g \text{ and } A x + B u + Ed \in X_{k-1} \} \) is the set of possible states, and

- \( D \) is the set of all possible demand.
1.3. DYNAMIC PROGRAMMING

The terminal state cost function \( J_0 \) and the corresponding set of feasible states \( \mathcal{X}_0 \) must be given as input. We assume that \( J_0 \) is linear (or piecewise affine convex) and that \( \mathcal{X}_0 \) is a polyhedral set. Thus, problem (1.5)-(1.6) is a parametric Linear Program where the state vector \( x(k) \) is the parameter.

From now on we will assume that there is no cost on the control variables, and the constraint (1.4) is of the form \( Fx(t) \leq g \). Furthermore we take \( F \) of the following form,

\[
\begin{pmatrix}
-I \\
I
\end{pmatrix}.
\]

(1.7)

When we have a certain number of time steps, an interesting thing to do is to actualize the monetary value. This can be done by introducing a discount factor. If the interest rate is \( r \geq 0 \) then it is easy to see that having \( S \) CHF in \( n \) time steps is the same as having \( (1 + i)^{-n} \cdot S \) today. The discount factor is \( \delta = (1 + i)^{-1} \). So, in the relation (2.5) above, we just need to introduce the discount factor in front of the term of the previous time step, that is \( J_{k-1} \). It becomes:

\[
J_k(x(k)) = \min_{u(k) \in U_k} \left\{ q^T x(k) + r^T u(k) + \delta \cdot \mathbb{E}\{J_{k-1}(A x(k) + B u(k) + d(k))\} \right\}.
\]

Therefore, choosing a discount factor \( \delta < 1 \) leads to convergence of the value of the objective function as \( N \) goes to infinity.

As a solution for this problem we have a value function \( J(x) \), a set \( \mathcal{R} \) of feasible regions. For each region there is a different cost vector which is as follows:

region \( i \): \( J_i(x) = c_i + \sum_j a_{ij} x_j \).

Note that as one can deduce from this formula the overall objective function is piecewise affine linear. We can write this for all the regions as the following:

\[ J(x) = Cx + c. \]

Once we have a state \( x \), the problem is to compute the expected case cost-to-go of this given state, because \( x \) belongs to only one region, or to none (if the state is not feasible). One method would be to see in which region belongs \( x \) and then take the objective function of this region and compute it. But this will be very long if we have a lot of regions. A better method consist to use the fact that the objective function is piecewise linear convex and therefore we have,

\[ J(x) = \max_i \{J_i(x)\}. \]

This can be easily understood in the case where we are only varying one component of \( x \). This can be seen at looking the Fig. 1.3, the solid line represent the objective function. We take every piece that is affine and extend them (dashed lines). Given a state \( x \), we easily see that \( J(x) = \max_i \{J_i(x)\} \) which is represented by the red point in the Fig 1.3.
1.4 INITIAL MODEL

We present in this part the initial model of this project. It is not complicated but it will easily allow the extensions of the model that we are willing to investigate later on. Consider a supply chain with one supplier $W$ that delivers goods to two retailers $X_1$ and $X_2$ through two independent paths. The shipping delays between $W$ and $X_1$ or $X_2$ is two time units, and the ordering delay is zero for both retailers and the time horizon is $N = 10$. The corresponding graph, after putting the necessary additional nodes, is represented in Fig. 1.4. The ordering cost is zero, the maximum order is 8 and the holding cost in each retailer is 0.5 per unit of material per unit of time. We consider that the demand coming to $X_1$ and $X_2$ takes the values 0 or 8 with same probabilities.

The matrices of the equation (1.3) are in this case:
1.4. INITIAL MODEL

\[
A = \begin{pmatrix}
1 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}, \quad B = \begin{pmatrix}
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 1 \\
0 & 0 \\
0 & 0
\end{pmatrix}, \quad E = \begin{pmatrix}
-1 & 0 \\
0 & -1 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0
\end{pmatrix}.
\]

We now set the following conditions on the inventory of each node of the supply chain:

\[
0 \leq x_i \leq 24 \quad \text{for } i = 1, 2,
\]
\[
0 \leq x_i \leq 8 \quad \text{for } i = 3, \ldots, 6.
\]

We can now determine the optimal control policy by solving the parametric LP, (1.5)-(1.6) recursively from \(k = 1\) to \(N\), for which we use the MPT toolbox [7]. Recalling that this solver finds an optimal policy to minimize the expected cost while fulfilling the demand, we find 4 regions. This is intuitive because we have two independent paths and thus \(2^2\) possibilities as shown in the following table:

<table>
<thead>
<tr>
<th></th>
<th>low inventory in path 1</th>
<th>sufficient inventory in path 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>low inventory in path 2</td>
<td>(X_1) and (X_2) order</td>
<td>(X_2) orders</td>
</tr>
<tr>
<td>sufficient inventory in path 2</td>
<td>(X_1) orders</td>
<td>no order</td>
</tr>
</tbody>
</table>

The feasible region of this problem is given by \(x_i \geq 0, \quad 1 \leq i \leq 6,\) and

\[
\begin{align*}
x_1 + x_3 & \geq 8, \\
x_1 + x_3 + x_4 & \geq 16, \\
x_1 + x_3 + x_4 & \leq 24, \\
x_2 + x_5 & \geq 8, \\
x_2 + x_5 + x_6 & \geq 16, \\
x_2 + x_5 + x_6 & \leq 24,
\end{align*}
\]

where the third and sixth constraints avoid the possibility that \(X_1\) and \(X_2\) have more inventory than what they are allowed to carry (i.e., 24), and the other constraints are here in order to be able to fulfill any demand at any time step.

In this simple model, the optimal control law is nothing other than the order-up-to policy also known as hedging point policy, which will ensure that the demand will be fulfilled in any case.

**Definition 4.** The hedging point level is the maximum demand per period times the time it takes from ordering to actually get what we ordered.

Given this, we can easily compute the hedging point level of the two retailers which is in both cases \(8 \cdot 3 = 24\). The resulting optimal control law is:
1.4. INITIAL MODEL

\[ u_1 = \max(24 - x_1 - x_3 - x_4, 0); \]
\[ u_2 = \max(24 - x_2 - x_5 - x_6, 0). \]

One can argue that there is no point in making such a model, the two paths are totally independent and thus it would be sufficient to look at a single path. The fact is that we will need this model with two paths in order to compare it with a model where we will introduce the option of transshipments between \( X_1 \) and \( X_2 \) later on.

**Conclusion**

We argued that the next state of the network depends on the current state, on the control input and on the disturbance, which in our case is the demand. To derive an optimal policy we will use the principle of dynamic programming, whose each step is formulated as a parametric LP, and which will give us an optimal control law that changes on each time step. In the following chapters we will be interested in such policies in extended model.

The extension of the model consists of the studies of real factors that can influence the evolution of the supply chain in time. The factors we will discuss concern correlated demand, backlogs and transshipments. In each subject we will see some theoretical tools that have helped us to understand the concept and introduce it in an adequate model that shows how such a modification in the supply chain can influence the processes that control the flows, i.e., the order policy.
Chapter 2
Correlated demand

The main question that has to be answered is how can temporal and locational demand correlations be integrated into the model. So we separate the study and consider in a first step the correlation due to time factors and then due to location factors. For the first type: on one hand it can be the reflect of the fact that the demand for one specific consumer product doesn’t change a lot day to day, for example butter is used everyday throughout the year; or that the demand can be extremely different depending on the season of the year (variation is due to seasonal variation but each summer it is similar to the past summer and each winter is similar to the past winter), one can think of ice cream sales which are higher in summer than in winter or heavy coats which has the opposite effect. But on the other hand, one can interpret this time correlation such as fashion products, for example clothes (or wearing a cap); but this last consideration is difficult to model because it’s difficult to know if the new fashion product will be liked or not. So unfortunately we will concentrate on the first hand, more precisely on the case of a product where the demand is approximatively the same all the year. In the second part of this section we concentrate on locational correlation, i.e., the correlation between the demand Retailer 1 faces and the one Retailer 2 faces.

2.1 Time correlation

For modeling time correlation, the first idea is to use time series models. A time series is a sequence of observation taken over time, for example, a sequence of daily demand of a product. By a time series model, we mean a model for a stochastic process. As we will see, in supply chains, time series models are very useful for forecasting demand. In fact there exist interesting books about time series models that incorporate seasonal effects. This subject is considered in [3]. Here, we will not cover this latter effect, so we take interest in the simple case where we consider no seasonal products. So we consider that the demand $D_t$ at each time step, has the same type of random behavior for all $t = 0, 1, \ldots$. To fully understand the following let us define some time series terminology:

**Definition 5.** A process is *weakly stationary* if its mean, variance, and covariance are un-
2.1. TIME CORRELATION

changed by time shifts. More precisely, \( Y_1, Y_2, \ldots \) is a weakly stationary process if:

- \( \mathbb{E}[Y_i] = \mu \) (a constant) \( \forall i \);
- \( \text{Var}[Y_i] = \sigma^2 \) (a constant) \( \forall i \); and
- \( \text{Corr}(Y_i, Y_j) = \rho(|i - j|) \) \( \forall i \) and \( \forall j \) for some function \( \rho(h) \).

**Definition 6.** The sequence \( Y_1, Y_2, \ldots \) is a weak white noise process with mean \( \mu \), denoted \( \text{WhiteNoise}(\mu, \sigma^2) \), if:

- \( \mathbb{E}[Y_i] = \mu \) (a constant) \( \forall i \);
- \( \text{Var}[Y_i] = \sigma^2 \) (a constant) \( \forall i \); and
- \( \text{Corr}(Y_i, Y_j) = 0 \) \( \forall i \neq j \).

We want the demand over time to be correlated so the weak white noise model does not fit, but it can be useful if we consider an AR(1) process which has the following form:

\[
D_t - \mu = \phi(D_{t-1} - \mu) + \epsilon_t,
\]

where \( \epsilon_1, \epsilon_2, \ldots \) is a white noise process with mean 0, that is \( \text{WhiteNoise}(0, \sigma^2) \) and where \( \mu \), the mean, and \( \phi \) are constant parameters. As \( \phi(D_{t-1} - \mu) \) is the "memory" of the past into the present value, and, as we suggest, that the demand on one day is completely determined by the previous plus a new piece of information represented by \( \epsilon_t \), we can choose to take \( \phi = 1 \). It can be shown that it is a non stationary process. We have:

\[
D_t = D_{t-1} + \epsilon_t,
\]

this is known as a random walk process. So supposing we start the process at an arbitrary point \( D_0 \), we can see that,

\[
D_t = D_0 + \epsilon_1 + \cdots + \epsilon_t.
\]

Then by the linearity of the expectation we have,

\[
\mathbb{E}[D_t] = \mathbb{E}[D_0] + \mathbb{E}[\epsilon_1] + \cdots + \mathbb{E}[\epsilon_t] = \mathbb{E}[D_0] \quad \forall t,
\]

where \( \mathbb{E}[D_0] = D_0 \) is a constant which depends entirely on the starting demand. Here we suppose this starting demand known by simply observing it. Moreover,

\[
\text{Var}[D_t] = \text{Var}[D_0] + \text{Var} \left[ \sum_{i=1}^{t} \epsilon_i \right] + 2 \text{Cov} \left( D_0, \sum_{i=1}^{t} \epsilon_i \right) \quad \forall t.
\]

We have:

- \( \text{Var} [D_0] = 0 \), because, as we said \( D_0 \) is a constant.
### 2.1. TIME CORRELATION

- \( \text{Var} \left[ \sum_{i=1}^{t} \epsilon_i \right] = \text{Var}[\epsilon_1] + \text{Var} \left[ \sum_{i=2}^{t} \epsilon_i \right] + 2\text{Cov} \left( \epsilon_1, \sum_{i=2}^{t} \epsilon_i \right) = \sum_{i=1}^{t} \text{Var}[\epsilon_i] = t\sigma^2 \), because of the bilinearity of the covariance and that the \( \epsilon_i \) are uncorrelated so \( \text{Cov}(\epsilon_i, \epsilon_j) = 0 \).

- \( \text{Cov} \left( D_0, \sum_{i=1}^{t} \epsilon_i \right) = \sum_{i=1}^{t} \text{Cov}(D_0, \epsilon_i) = 0 \), because \( D_0 \) and \( \epsilon_i \) are independent \( \forall i \).

Then, \( \text{Var}[D_t] = t\sigma^2 \).

In our model we will use the idea mentioned in the previous part: the random walk process as describing the process \( D_t \) of the demand. But in order to be consistent with the discreteness of the model we should introduce some modifications to it. Instead of having a WhiteNoise \( \epsilon_t \) as the new information we will model it by \( e(t) \) which can take either the value 1 or \(-1\) with same probability. So we have the following dynamics for the demand:

\[
D_t = D_{t-1} + e(t), \quad \text{with} \quad e(t) \in \{-1, 1\}.
\]

The question is now how to integrate this idea into the model. The first possibility was to add an additional node \( X_h \) which "remembers" the history of the demand (in our case it remembers only the last one). That is,

\[
x_h(t) = D_{t-1}.
\]

Then, based on the first chapter, the dynamics of a shop denoted by \( X_1 \) can be written

\[
x_1(t+1) = z_{in}(t) - (x_h(t) + e(t)).
\]  \hfill (2.1)

In order to get the reader familiar with what we just introduced above we now give a simple example. Suppose we have one shop \( X_1 \) with one supplier \( W \), and there is no time delay between the time where amount \( u \) we order and the delivery from \( W \), see Fig. 2.1.

![Figure 2.1: Network with additional state variables to time correlation model](image)

With respect to the first chapter we also give the dynamics of this model:

\[
x(t+1) = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} x(t) + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} -1 \\ 1 \end{pmatrix} e(t),
\]

where \( x(t) = \begin{pmatrix} x_1(t) \\ x_h(t) \end{pmatrix}, u(t) = u(t) \) and \( e(t) = e(t) \).
2.1. TIME CORRELATION

Remark 4. Contrary to the explanation of the first chapter concerning the dynamics, here the demand is not \( e(t) \), but as suggests (2.1), it is \( x^h(t) + e(t) \).

So far this model is a good representation of the reality, but it is not suited for our modeling framework, which assumes that the state space is bounded for any arbitrary time horizon. Here, as the term \( e(t) \) is either \(-1\) or \(1\), it is possible that after a finite number of time steps, \( x^h(t) \) which is \( x^h(t - 1) + e(t) \), exceeds the boundary given. To illustrate this point we give a small example.

Example 2. If \( N \), the number of time steps is 6 and that \( x^h \) lies between 0 and 15, then it will work and the possible values of \( x^h(0) \) are \{6; 7; 8; 9\}. But in the case of \( N = 10 \) there will be no feasible initial state. This is because for any value of \( x^h(0) \) at least one of the boundaries (here 0 and 15) can be exceeded in less than ten time steps.

We will therefore introduce a new model, which is less realistic but much more comfortable to work with. Instead of having \( e(t) \in \{-1; 1\} \) we will take \( e(t) \in \{0; 1\} \) and the additional node \( X^h \) will not recall the past demand, but rather the last new information. Therefore we can write:

\[
x^h(t) = e(t - 1).
\]

The demand will be, by definition of this model:

\[
D_t = e(t) + x^h(t).
\] (2.2)

One can see from (2.2) that the demand on each day is either 0, 1 or 2. We can be interested in knowing the probability that the demand is \( x \) on one day given that it was \( y \) on the previous day, i.e, \( P(D_{t+1} = x | D_t = y) \). The equation (2.2) above can be rewritten as

\[
D_t = e(t) + e(t - 1).
\] (2.3)

From this equation we can see that if the demand is 2 on day \( t \) this implies that \( e(t) = 1 \) and hence

\[
D_{t+1} = e(t + 1) + e(t) = \begin{cases} 1 & \text{if } e(t + 1) = 0 \text{ and } e(t) = 1 \\ 2 & \text{if } e(t + 1) = 1 \text{ and } e(t) = 1 \end{cases}
\]

As \( e(t + 1) \) takes the values \( \{0, 1\} \) with equal probability, the probability that \( D_{t+1} \) is 1 or 2, given that \( D_t = 2 \), is the same. Overall, the demand process \( D_t \) is generated by the Markov chain given in Fig. 2.2.

Let us be more clear in why the probabilities of having the demand \( D_{t+1} \) given that \( D_t = 1 \) are the ones given in Fig. 2.2. The meaning of \( D_t = 1 \) is that we either have \( X^h(t) = 1 \) and \( e(t) = 0 \) (first case) or \( x^h(t) = 0 \) and \( e(t) = 1 \) (second case). If we look at the two cases:

In the first one, we have:

\[
D_{t+1} = \begin{cases} 0 & \text{if } e(t + 1) = 0 \\ 1 & \text{if } e(t + 1) = 1 \end{cases}
\]
2.1. TIME CORRELATION

In the second one this becomes:

\[ D_{t+1} = \begin{cases} 
1 & \text{if } e(t+1) = 0 \\
2 & \text{if } e(t+1) = 1 
\end{cases} \]

We get that in half of the case \( D_{t+1} = 1 \), that is why \( P(D_{t+1} = 1 | D_t = 1) = 0.5 \). And by the fact that \( e(t) \in \{0, 1\} \) with same probabilities we have \( P(D_{t+1} = 0 | D_t = 1) = P(D_{t+1} = 2 | D_t = 1) = 1/4 \). Note that the equation (2.3) can be rewritten as:

\[ D_t = e(t) - \theta e(t-1) \]

with \( \theta = -1 \) which is (in the case where \( \epsilon_t \in \text{WhiteNoise}(0, \sigma^2) \)) very similar to a process known as a moving average process or \( MA(1) \) very famous in financial mathematics. It is a good process to model correlation at only short lags (in our case, only at lag 1). The moving average process is a weighted average of the past values of the terms \( e(t) \), rather than of past value \( D_t \) as our previous model.

Our goal is now to use the time correlation in our initial model. Using the same settings, we add two "memory" nodes, one for each shop; \( x^h_1(t) \) for the disturbance \( e_1(t-1) \) that comes to \( X_1 \) at time \( t \) and \( x^h_2(t) \) for the disturbance \( e_2(t-1) \) that comes to \( X_2 \) at time \( t \). The graph of the supply chain will therefore look like the graph on Fig. 2.3. Here we explain how we can express this model extension via the problem data of the general stochastic dynamic programming problem introduced in Section 1.3. As we are adding 2 new nodes, the total number of nodes is now 8. So \( F \) becomes a \( 2 \cdot 8 \times 8 \) matrix, but still of the form given in (1.3). And \( g \) becomes in our case \( (0, 0, 0, 0, 0, 0, 6, 6, 2, 2, 2, 2, 1, 1)^T \). For \( A, B \) and \( E \) we can generalize how to modify them. Let be \( n_s \) the number of shops we own. Then we add \( n_s \) artificial "memory" nodes, one for each shops. So we define the state of the system as

\[ x(t) = (\overbrace{x_1(t), \ldots, x_{n_s}(t)}^{\text{nodes relative to the } n_s \text{ shops}}, \overbrace{x_{n_s+1}(t), \ldots, x^h_1(t), \ldots, x^h_{n_s}(t)}^{\text{transient nodes}}, \overbrace{x^h_1(t), \ldots, x^h_{n_s}(t)}^{\text{memory nodes}}) , \]

we can rewrite the equation (1.2) as

\[ x(t+1) = \begin{pmatrix} A & -I_{n_s} \\ 0 & 0 \end{pmatrix} x(t) + \begin{pmatrix} B \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} -I_{n_s} \\ 0 \end{pmatrix} e(t) . \]
2.1. **TIME CORRELATION**

![Diagram of the initial model with time correlation](image)

Figure 2.3: Initial model with time correlation.

Applying this idea to our case we have the following:

\[
A = \begin{pmatrix}
1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & -1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{pmatrix},
B = \begin{pmatrix}
0 & 0 \\
0 & 0 \\
0 & 0 \\
1 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
\end{pmatrix},
E = \begin{pmatrix}
-1 & 0 \\
0 & -1 \\
0 & 0 \\
0 & 0 \\
1 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
\end{pmatrix},
\]

where the demand takes the following values with same probabilities,

\[
D_t \in \{(0,0), (0,1), (1,0), (1,1), (1,2), (2,1), (0,2), (2,0), (2,2)\},
\]

(where \((i,j)\) means a demand of \(i\) to shop 1 and a demand of \(j\) to shop 2). The solution is in this case, for a time horizon of 10 time steps, the following feasible region:

1. \(x_1^h \leq x_1 + x_3\),
2. \(3 + x_1^h \leq x_1 + x_3 + x_4 \leq 6 + x_1^h\),
3. \(x_2^h \leq x_2 + x_5\),
4. \(3 + x_2^h \leq x_2 + x_5 + x_6 \leq 6 + x_2^h\).

As we have already said, the two paths are independent while there are no transshipments between them, therefore we need to analyze what is happening only in one path, say the
2.1. TIME CORRELATION

upward path. The first condition above is given in order for the demand on the next stage to be completely fulfilled. By analyzing different parts of the second condition we will try to characterize the optimal order policy.

The purpose is to find whether it is useful to order at this stage or not and if yes, how much to order. To find what happens in this situation consider that the inventory of $X_1$ at time 0 is $x_1$, the one of $X_3$ is $x_3$ and the one of $X_4$ is $x_4$. Denote by $x_1^h(i)$ the contents of $X_1^h$ at stage $i$. The following discussion can be read in parallel with the graph on Fig. 2.4.

First of all note that, as we want to satisfy any demand, we are doing as if the maximum demand possible is reached in each stage given the setup (worst case). The first demand is then either 1 or 2 depending on $x_1^h(0)$ (which can be 0 or 1) and as $x_1^h(i) = 1$ for $i = 2, 3$ then $D_i = 2$ for $i = 2, 3$. As the order made at stage 0 is relevant for satisfying the demand $D_3$ it may be useful to see what happens between stages 2 and 3. This is summarized in the graph on Fig. 2.4.

Figure 2.4: Evolution of the inventory of $X_1$ throughout time

Now what happens in the third stage. As we want to fulfill $D_3$ we need to have

$$2 \leq x_1 - (1 + x_1^h(0)) + x_3 - 2 + x_4 + u_1;$$
2.2. LOCATION CORRELATION

which gives us

\[ 5 + x_1^h(0) \leq x_1 + x_3 + x_4 + u_1. \]

So, should we order at time 0 or not? This will depend on whether \( x_1 + x_3 + x_4 \geq 5 + x_1^h(0) \) or not. If it is the case, then \( u_1 \) could be zero, and it will be zero as we don’t want unnecessary holding costs. But in the case where the inequality is not satisfied we should order and the order should be

\[ u_1 = 5 + x_1^h(0) - (x_1 + x_3 + x_4). \]

Remark 5. Note that the orders at stage 1 and 2 and so on are made using the same result based on the state of the supply chain at stage 1, 2 an so on, respectively.

The transition matrix of the demands for the last model above is given below. The probabilities are based on the ones found for the simpler model and summarized in the Markov Chain in Fig. 2.2. \((i, j)\) means that the retailer \( X_1 \) faces a demand of \( i \) units and the retailer \( X_2 \) faces a demand of \( j \) units.

\[
\begin{pmatrix}
(0,0) & (1,0) & (0,1) & (1,1) & (2,0) & (2,1) & (1,2) & (2,2) \\
(0,0) & 1/4 & 1/4 & 1/4 & 1/4 & 0 & 0 & 0 \\
(1,0) & 1/8 & 1/4 & 1/8 & 1/4 & 1/8 & 0 & 0 \\
(0,1) & 1/8 & 1/8 & 1/4 & 1/4 & 0 & 1/8 & 0 \\
(1,1) & 1/16 & 1/8 & 1/8 & 1/4 & 1/16 & 1/8 & 1/8 \\
(2,0) & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 \\
(0,2) & 0 & 0 & 1/4 & 1/4 & 0 & 1/4 & 0 \\
(2,1) & 0 & 1/8 & 0 & 1/4 & 1/8 & 0 & 1/8 \\
(1,2) & 0 & 0 & 1/8 & 1/4 & 0 & 1/8 & 1/8 \\
(2,2) & 0 & 0 & 0 & 1/4 & 0 & 1/4 & 1/4 \\
\end{pmatrix}
\]

2.2 Location Correlation

We have seen the case where the demand that a retailer faces can vary in time (time series). In this section we ask ourselves another question. Given a company that consists of one common supplier and two retailers, will the demand that the first retailer faces be correlated with the demand the second retailer faces?

Reality often shows us that this is indeed the case. Take a company that specializes in fashionable goods for example and assume it operates with two stores. The demand for those goods is often influenced by trends and/or reputation of the company. Let’s say the company comes up with a new collection that eventually becomes successful. The demand for those goods will rise in general, so if we see an increase of the demand in store 1 then there’s a good chance that this will be the same in store 2. Trend and reputation of a company therefore account for positive correlations between the two demands.

The demands could, however, be negatively correlated. The most common example is a chain of supermarkets operating in the same neighborhood. We will assume that there is no general preference for any of the stores and that the neighborhood’s inhabitants randomly
2.2. LOCATION CORRELATION

choose to which store to go. It is therefore logical to assume that if a person goes to supermarket 1 on day 1 that he will not go to store 2 the same day. This means that while store 1 gains one customer, store 2 loses one. At the end of the day we’ll have three groups of people: the people that went to store 1, the people that went to store 2 and those that chose not to go buy groceries that day.

We can look at this problem in a different way: from the shopper’s perspective. There are 3 choices:

1. go to store 1,
2. go to store 2, or
3. stay home.

Let us assume we can determine the probabilities for each choice. We would like to construct a model that would determine the joint distribution function of the two demands, which we model as two random variables $D_1$ and $D_2$. An appropriate joint distribution function that fits such a model is the multinomial distribution function. Let us introduce the following notations:

- probability that a person goes to store 1: $p_1$.
- probability that a person goes to store 2: $p_2$.

Let us assume that the neighborhood’s population is $n$. Then the probability distribution function (p.d.f) of the multinomial distribution is

$$f(d_1, d_2) = \mathbb{P}(D_1 = 1, D_2 = d_2) = \frac{n!}{d_1!d_2!(n - d_1 - d_2)!} p_1^{d_1} p_2^{d_2} (1 - p_1 - p_2)^{n-d_1-d_2}.$$ 

The variance/covariance matrix is

$$\Omega = \begin{pmatrix} \text{Var}(D_1) & \text{Cov}(D_1, D_2) \\ \text{Cov}(D_1, D_2) & \text{Var}(D_2) \end{pmatrix} = \begin{pmatrix} np_1(1-p_1) & -np_1p_2 \\ -np_1p_2 & np_2(1-p_2) \end{pmatrix}.$$ 

So then $\text{Corr}(D_1, D_2) = \rho = -\sqrt{\frac{p_1p_2}{(1-p_1)(1-p_2)}}$.

So we can look at this in two ways, either we know $p_1$ and $p_2$, we then easily get $\rho$ or we know $\rho$ in advance (measured empirically by the company for example). We’ll consider the latter.

Say we know $\rho$, $p_1$ and $D_1$. What is the expected demand in store 2? In order to do this we need to calculate $\mathbb{E}[D_2|D_1 = d_1]$. Let’s assume that $\mathcal{D}$ is the set of all possible values of $D_2$.

The computation is:

$$\mathbb{E}[D_2|D_1 = d_1] = \sum_{d_2 \in \mathcal{D}} \frac{f_{D_1, D_2}(d_1, d_2)}{f_{D_1}(d_1)} \cdot d_2.$$ 

22
2.2. LOCATION CORRELATION

\[
\sum_{d_2 \in D} p_{d_2}^2 (1 - p_1 - p_2)^{n-d_1-d_2} \cdot \frac{n!}{d_1!(n-d_1-d_2)!} \cdot \frac{(1-p_1)^{n-d_1}}{(n-d_1)!} = \sum_{d_2 \in D} \frac{p_{d_2}^2 (1 - p_1 - p_2)^{n-d_1-d_2}}{d_2!(n-d_1-d_2)!} \cdot \frac{(1-p_1)^{n-d_1}}{(n-d_1)!} \cdot d_2.
\]

Example Let us take a simple population: \( n = 2 \) with \( p_1 = 1/3 \) \( \rho = -1/2 \) a simple computation gives us \( p_2 = 1/3 \). Store 1 reports a demand of 1. What is the expected demand in store 2?

\[
\mathbb{E}[D_2|D_1 = 1] = \sum_{d_2 \in D} \frac{(1/3)^{d_2}(1/3)^{1-d_2}}{d_2!(1-d_2)!(2/3)}
\]

\[
= 0 \cdot \frac{(1/3)^0(1/3)^2}{2/3} + 1 \cdot \frac{(1/3)^1(1/3)^0}{2/3} = 1/2.
\]

Therefore the expected demand in store 2 is \( \mathbb{E}[D_2|D_1 = 1] = 1/2 \).

The joint distribution is

| \( d_1 \) | 0 | 1 | 0 | 1 | 2 | 0 |
| \( d_2 \) | 0 | 0 | 1 | 1 | 0 | 2 |
| \( f(d_1,d_2) \) | 1/9 | 2/9 | 2/9 | 2/9 | 1/9 | 1/9 |

Model/Problem data

Recall that:

\[
x(t + 1) = Ax(t) + Bu(t) + Ed(t).
\]

The model is the same as in the initial model, i.e.,

\[
A = \begin{pmatrix}
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix},
B = \begin{pmatrix}
0 & 0 \\
0 & 0 \\
0 & 1 \\
0 & 0 \\
0 & 0 \\
0 & 1
\end{pmatrix},
E = \begin{pmatrix}
-1 & 0 \\
0 & -1 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0
\end{pmatrix}.
\]
2.2. LOCATION CORRELATION

However the demand is not 0 or 8 but takes 0, 1, 2 with the probabilities shown in the table.

The regions and the control laws are exactly the same as in the initial model, therefore locational correlation does not change the control law. This is without doubt due to the fact that the two stores are not interconnected. The two stores order from the supplier independently and do not adjust their control policy according to the changes in demand in the other store. In other words, the "order-up-to" policy still seems to be optimal.

An interesting question would be: would there be any effects on the order policy if transshipping was allowed? We will find in Chapter 5 that this is indeed the case.

Conclusion In the case of a time correlated demand the policy does not depend only on the current state but also directly on the previous demand. This is explained by the fact that the demand is correlated in time, therefore we have some information about the upcoming demand and hence we can apply a better policy. In the case of location correlation, the policy does not differ from the initial model because there are no transshipments. And hence it is not useful to know what is the demand in the other shop as we can do nothing with this information.
Chapter 3

Backlogs

This third chapter is about considering the possibility to leave demand partially unfilled, and paying instead a penalty per unit and time step. This penalty or the so-called backlog cost can be due to the fact that the shop has to get supplied by another shop which belongs to a competitor and which can supply it directly without time delay. We will consider the case where we pay a penalty fine per unsatisfied demand per time step. (Thus we can precise that this partially unfilled demand happens when the demand hasn’t been satisfied on time or simply because it is less expensive to reimburse the client than to satisfy him.)

This concept of backlogs is, like bookings and billings, a critical element in the delivery of predictable financial results and reliable customer service (for a company). Here we try to integrate this important concept into a very simple model where we have only one shop with one supplier without time delay. Then we will introduce it on our initial model and explain where changes come into our setup. However, before doing that two points will be presented. First, in order to illustrate shortages (i.e., a lack of product) we will talk about a continuous deterministic model with uniform demand "with continuous check-up of stock" where shortages are allowed. Second, we will introduce, with the help of stochastic dynamic programming, a case where the demand $D$ is a discrete random variable with known distribution, and where the backlogs costs are modeled by a shortage cost per unit.

3.1 Continuous deterministic model

The aim of this model is to illustrate the backlogs, and how it can modify an ordering policy. At this level some hypotheses have to be made, we have:

- only one type of article, for example ice cream or whatever,
- a constant demand of $a$ articles/unit of time,
- a cost of production of $x$ articles as

$$C(x) = \begin{cases} 
0 & \text{if } x = 0 \\
K + cx & \text{if } x > 0 
\end{cases}$$
3.1. CONTINUOUS DETERMINISTIC MODEL

where $K$ is a fixed cost for launching a production and $c$ the cost to produce one unit of the good,

- a holding cost of $h$ per article and unit of time,
- an infinite supply.

1. First we don’t allow shortages in order to derive an optimal policy which will be compared with the one where shortages are allowed. The first things that we have to compute is the cost of one cycle. For that we need to compute the holding cost which depends on the level of the stock. At the beginning of the cycle we order a certain amount of articles $Q$, so we have a stock of $Q$. The constant demand decreases the level of the stock with a slope of $-a$, see on the FIG 3.1 to have a better view of the development of the inventory. Thus we can determine the holding cost for one cycle which is the cost of holding one article times the level of the stock $s(t)$:

$$h \int_0^Q s(t) \, dt = h \int_0^Q (Q - at) \, dt = hQ^2 \frac{2}{2a}.$$

![Figure 3.1: Level of the stock in function of the time](image)

Having the holding cost, we can determine the cost of one cycle as

$$K + cQ + hQ^2 \frac{2}{2a},$$

and so the cost per unit of time, dividing this expression by the length of one cycle $Q/a$, we obtain

$$\frac{K + cQ + hQ^2 \frac{2}{2a}}{Q/a} = \frac{aK}{Q} + ac + hQ \frac{2}{2}.$$  \hspace{1cm} (3.1)
3.1. CONTINUOUS DETERMINISTIC MODEL

Now we can find an optimal level of stock, \( Q_{\text{opt}} \), which minimizes the cost of one cycle. Then we have an optimal control policy, which is ordering \( Q_{\text{opt}} \) for each beginning of cycle. To find \( Q_{\text{opt}} \), we differentiate with respect to \( Q \) the above formula (3.1). Thus the optimal value \( Q_{\text{opt}} \) has to satisfy

\[
- \frac{aK}{Q_{\text{opt}}^2} + \frac{h}{2} = 0 \Rightarrow Q_{\text{opt}} = \sqrt{\frac{2aK}{h}}.
\]

This result is known as the Wilson formula, see [5] for more details.

2. In this step we allow shortages so we can add the following hypotheses:

- we have a shortage cost of \( p \) per article and unit of time.

As in the first case, at the beginning of a cycle we order a certain amount \( Q \), but we can have a negative level of stock. So, denoting by \( S \) the level of the stock after the ordering, we have a possible value of shortage of \( Q - S \). The constant demand decreases the level of the stock with a slope of \( -a \), see on the FIG 3.2. In the same way as in the first case, we derive the holding cost per cycle

\[
h \int_{0}^{\frac{S}{a}} s(t)dt = h \int_{0}^{\frac{S}{a}} (S - at)dt = h \frac{S^2}{2a},
\]

and the shortage cost per cycle

\[
p \int_{\frac{S}{a}}^{\frac{Q}{a}} (-s(t))dt = h \int_{\frac{S}{a}}^{\frac{Q}{a}} (at - S)dt = p \frac{(Q - S)^2}{2a}.
\]

level of the stock

\[\begin{align*}
S & \quad \frac{S}{a} \\
0 & \quad \frac{Q}{a} \\
Q - S & \quad t
\end{align*}\]

Figure 3.2: Level of the stock in function of the time

27
3.2. **DISCRETE MODEL**

Thus the total cost of one cycle is:

\[ K + cQ + h\frac{S^2}{2a} + p\frac{(Q-S)^2}{2a}, \]

and then dividing by \( Q/a \), we obtain the total cost per unit of time:

\[ \frac{K + cQ + h\frac{S^2}{2a} + p\frac{(Q-S)^2}{2a}}{Q/a} = f(S, Q). \]

Now we can find an optimal level of stock, \( S_{opt} \), and the optimal ordering policy \( Q_{opt} \). \( S_{opt} \) and \( Q_{opt} \) have to satisfy, respectively,

\[ \frac{\partial f}{\partial S} = 0 \quad \text{and} \quad \frac{\partial f}{\partial Q} = 0, \]

then, after some algebra,

\[ S_{opt} = \sqrt{\frac{2aK}{h}}\sqrt{\frac{p+h}{p}} \quad \text{and} \quad Q_{opt} = \sqrt{\frac{2aK}{h}}\sqrt{\frac{p}{p+h}}. \]

**Remark 6.** We can observe that in the above formula of \( S_{opt} \) and \( Q_{opt} \), if \( p \) goes to infinity (i.e., penalty fine so expensive that we absolutely want to avoid it), \( S_{opt} \) tends to 0 and we find again the Wilson formula for \( Q_{opt} \).

### 3.2 Discrete model

Here we are interested in a more realistic case where the demand is not known, and where the goal is still to minimize the cost which in this case is now a random variable. Nevertheless, to simplify the study we need to make some assumptions:

- no fixed cost for launching a production,
- \( D \in \mathcal{D} \) represents the demand which is a discrete random variable, and \( \mathcal{D} \) is the set of values that \( D \) can take (the assumption here is that the demand doesn’t depend on the time \( t \) and we suppose to know its distribution, which is acceptable because it can be empirically estimated by previous studies on the market),
- \( p > 0 \) is the shortage cost per unit (the assumption here is that this cost doesn’t depend on the time period),
- \( h > 0 \) is the holding cost per unit (same assumption as for \( p \)),
- \( x(t) \) is the stock at time \( t \),
- \( u(t) \) is the quantity ordered at time \( t \),
3.2. DISCRETE MODEL

- We don’t integrate the buying price and the selling price because they are not supposed to be minimized.

The main idea is to express the cost of the eventual excess or lack of product we sell we are confronted of, and to minimize it. As this cost depends on the value that can take the demand, it is a random variable, so we attempt to minimize the expected cost. The cost at time $t$ is a function of the demand $D$, the order $u(t)$ and the stock $x(t)$:

$$C_t(D, u(t), x(t)) = \begin{cases} (D - (x(t) + u(t))) \cdot p & \text{if } x(t) + u(t) \leq D \\ (x(t) + u(t) - D) \cdot h & \text{if } x(t) + u(t) > D \end{cases}$$

In other words,

$$C_t(D, u(t), x(t)) = p \cdot \max\{D - (x(t) + u(t)), 0\} + h \cdot \max\{x(t) + u(t) - D, 0\}$$

So, denoting:

- $D_+(t) = \{d \in D : d \geq x(t) + u(t)\}$ and,
- $D_-(t) = D \setminus D_+(t)$,

we can write the expectation of the cost:

$$\mathbb{E}(C_t(D, u(t), x(t))) = p \cdot \sum_{d \in D_+(t)} (d - (x(t) + u(t))) \mathbb{P}(D = d) + h \cdot \sum_{d \in D_-(t)} (x(t) + u(t) - d) \mathbb{P}(D = d).$$

We want to minimize this expectation of the cost over a given time horizon $N$, so we have to minimize it for each time step, i.e.,

$$\min_{u(t)} \mathbb{E}(C_t(D, u(t), x(t))) , \text{ for } t = 1, \ldots, N.$$

This problem can be solved with the help of stochastic dynamic programming introduced in section 1.3, using the following recursion relation,

$$J_k(x(k)) = \min_{u(k)} \left\{ p \cdot \sum_{D \in D_+} \left[ (d - (x(k) + u(k))) + J_{k-1}(d - (x(k) + u(k))) \right] \mathbb{P}(D = d) + h \cdot \sum_{D \in D_-} \left[ (x(k) + u(k) - d) + J_{k-1}(x(k) + u(k) - d) \right] \mathbb{P}(D = d) \right\}.$$

**Remark 7.** If we try to apply this relation with given numbers, we can see that the computations by hand are very tedious. An important things is to be very careful in which domain, $D_+$ or $D_-$, $D$ is, depending on the value of $x(t)$ and $u(t)$. Moreover the calculation has to be made for many levels of stock $x(t)$ at time $t$, for each time step (because $D$ is random).

### 3.2.1 Modeling the problem

Before integrating this concept of backlogs to our initial model we will integrate it in a very simple case, to see and explain how we have to choose the input data to our general stochastic dynamic programming problem.
3.2. DISCRETE MODEL

Simple case

So let us consider the following simple model: suppose we have one shop $X$ with one supplier $W$, and there is no time delay between the time we make the order $u_1$ and the delivery from $W$. As an input in $X$, we have a random demand $D$ taking either value 0 or 8 with same probabilities. We have the following directed connected graph:

![Figure 3.3: Graph of the simple case](image)

Using the general problem described in the first chapter, the global dynamics of this model can be written as:

$$x(t + 1) = x(t) + u(t) - d(t)$$

where $x(t) \in \mathbb{R}$ denotes the state of the system, $u(t) \in \mathbb{R}$ the control input, and $d(t)$ the uncertain demand (which takes values 0 or 8 with same probabilities), at time $t$. Using (1.3) these settings can easily formulated using $A = 1$ and $B = 1$, and with the particularity that the stock $x$ can be negative. Allowing this, we have to change the infimum of $x$ and this is done by, as mentioned in section 1.3, changing $g$. It’s not 0 anymore but a positive number, let us say $x_{inf}$, so we allow the stock to be between $-x_{inf}$ and $x_{sup}$:

$$g = [x_{inf}, x_{sup}]^T.$$  

Then, to integrate the backlogs cost we have to change the $q$ vector into a matrix $Q$ which will manage the state cost. In this matrix is incorporated the holding cost $h$ and the shortage cost $p$. In our simple example it is just

$$Q = \begin{pmatrix} h \\ -p \end{pmatrix}.$$  

In order to make the stochastic dynamic programing we have to take the $\max_i \{Q_i x\}$. So if we have a negative stock, we consider a cost of $-xp$, which is non-negative because of the negativity of $x$, indeed $-xp > xh$. If we have a positive stock, we consider a cost of $xh$. In fact we do that it in such a way as to linearize the problem in order to formulate it as a parametric linear program. On the FIG. 3.4 we illustrate this idea.

In fact this little model without time delays is not very interesting in itself for the studies of backlogs, because, we can see that the absence of time delays allows us to keep a stock of 8 products at each step and thus satisfy the demand, which is the best policy, at least if the holding cost is smaller than the shortage cost. But, as we have seen above this simple model is helpful to know what and how we have to change the elements in the model.

After understanding this simple model, we can generalize how to change the $Q$ matrix to integrate the backlogs in a more sophisticated model. The main thing that has to be clear
is that we take \( \max_i \{ Q_i x \} \) to express the state cost. So if we have many shops, we have to write the \( Q \) matrix in a such way that the model takes in account all the possible cases of lack or excess of products in these shops. We model this by writing each case in one row of \( Q \), thus by taking \( \max_i \{ Q_i x \} \) the model considers the good state and we are able to compute the cost of the state. Thus we can apply this idea in our initial model, where the state cost of the additional nodes is 0. We have two shops, so denoting \( x_1 \) the level of the stock of the first shop and \( x_2 \) the level of the stock of the second shop, four cases can happen:

- \( x_1 > 0, x_2 > 0 \): so the corresponding row in \( Q \) is \( Q_1 = [h \ h \ 0 \cdots 0] \)
- \( x_1 > 0, x_2 < 0 \) \( \Rightarrow Q_2 = [h \ -p \ 0 \cdots 0] \)
- \( x_1 < 0, x_2 > 0 \) \( \Rightarrow Q_3 = [-p \ h \ 0 \cdots 0] \)
- \( x_1 < 0, x_2 < 0 \) \( \Rightarrow Q_4 = [-p \ -p \ 0 \cdots 0] \)

So if we have \( n \) shops we have \( 2^n \) possibilities and then \( 2^n \) rows to fill in \( Q \!^! \). The other things to change in the model are, as mentioned above and explained in Section 1.3, the constraint vector \( g \). In the same way as for the simple example above, we fill in our bounds for the stock of our shops adequately in \( g \).

**Initial model case**

In this part we take the initial model defined in Section 1.4 and add to it the possibility of leaving the demand partially unfilled. We have to make the modifications introduced above. So if we suppose that we allow at most 24 of the demand to be unsatisfied and recalling
3.2. DISCRETE MODEL

\( Fx \leq g \) where \( F \) is of the form \( \begin{pmatrix} -I \\ I \end{pmatrix} \), the state constraint becomes,

\[ g = (24, 24, 0, 0, 0, 24, 8, 8, 8, 8)^T. \]

Then if we suppose that the holding cost \( h \) is 0.5 and that the shortage cost \( p \) is 1, the \( Q \) matrix is:

\[
Q = \begin{pmatrix}
0.5 & 0.5 & 0 & 0 & 0 & 0 \\
0.5 & -1 & 0 & 0 & 0 & 0 \\
-1 & 0.5 & 0 & 0 & 0 & 0 \\
-1 & -1 & 0 & 0 & 0 & 0
\end{pmatrix}
\]

After applying the dynamic programing recursion the parametric LP solver from the MPT toolbox [7] with what we changed above, we obtain a solution with 25 regions. Each of these 25 regions has a different objective function. We are interested in interpreting these regions in order to derive a global policy. First of all we state the set of feasible states:

\[
-8 \leq x_1 + x_3 + x_4 \leq 24; \\
-8 \leq x_2 + x_5 + x_6 \leq 24.
\]

Like we have seen in Remark 3 of Chapter 1, we can focus only on one of the two paths because of their independence. So we can consider the path constituted by \( X_1, X_3 \) and \( X_4 \). With respect to the inequality above we can draw a real line that goes from \(-8\) to \(24\) for the sum of the goods that are in this path. Now let us divide this line in three parts as shown in Fig. 3.5.

![Figure 3.5: Total inventory in the path](image)

Remember that we are allowing backlogs. Nevertheless in our case, as the backlog cost is bigger than the holding cost, we may prefer to order a lot and then probably face a holding cost instead of gambling on low demand and ordering less and thus face probable shortage costs if the demand has been underestimated.

To simplify the notation, let us define \( x_p = x_1 + x_3 + x_4 \), the number of units of goods that is in the path. For the first policy, note that if the inventory in the path is between \(-8\) and \(8\) then there is not sufficient material to satisfy a big demand in the following two time steps. We would logically order \(16\) minus \(x_p\) (as a reference to a hedging point policy). But an order of \(16 - x_p \geq 8\) is not possible because of the constraints on the order \((0 \leq u_1 \leq 8)\). Therefore one would just simply order the maximum one can, that is 8. The second policy takes place in the case where \(x_p\) is between \(8\) and \(16\). Here one just has to order \(16\) minus what is in the path. Mathematically that is \(8 - x_p\). The last policy appears when \(x_p\) is bigger than \(16\). In this case the inventory of the path is sufficient to satisfy the maximum
3.2. DISCRETE MODEL

demand (which is 8 in our example) during two time steps. Therefore we will not order any material now and, depending on what might happen at the next time step (i.e., depending on the interval $x_p$ belongs to), maybe order then.

At this point we have three policies for one path, but remember that the regions are defined by the objective values which are the expected cost, and not by the policies. It can be that two regions which have, by definition, two different objectives values, have the same policy. That is exactly what we have here.

Indeed it appears in the case where the policy 1 and 2 are applied, that $x_1$ could either be positive or negative because of backlogs. Each case will give a different objective value because the shortage cost is different from the holding cost. It is not the case when the third policy is applied because we cannot have $x_p \geq 16$ and $x_1 \leq 0$ at the same time (because of the condition on $x_3$ and $x_4$).

At the end we have, for this single path, three policies and five regions. Because there are two independent paths, we therefore have a total of $3^2 = 9$ policies and $5^2 = 25$ regions.

Conclusion

We have seen that in the continuous time case with known demand, depending on the holding and shortage cost, the optimal policy will sometimes allow demand to be unfulfilled, in other words the states where the inventory is "negative" are recurrent states. This proved to be true in our example too. Logically, the penalty cost has a big influence in this analysis.
Chapter 4

Transshipments

Up until now we used a fairly simple model for our distribution firm. In this section we will extend this model by introducing the option of transshipments. What we mean by transshipments is that the two retailers can now exchange goods between themselves as opposed to only being able to get goods from the supplier. Our goal in this section is to find what are the advantages of such an option and to evaluate its value.

The conjecture is that the transshipments option could allow to work on lower joint inventory and thus save part of holding cost. For example if a retailer is experiencing difficulties in satisfying demand can now get help from the other retailer, assuming that the other retailer is in a position to help (i.e., has surplus inventory). This is indeed an advantage because it is logical to assume that the time delay of shipping goods between the two retailers is shorter than if we ordered from the supplier. A good example would be two retailers operating in the same city but getting supplied from another city. It is obvious that transporting goods from one to another takes less time than ordering and getting supplied from the supplier.

What this means mathematically is that the domain of feasible states is now larger. In the case where the maximum demand is 8, we could now have a feasible state where $x_2 = 6$. 

![Figure 4.1: Model with transshipments](image_url)
This could happen in the case where $x_1 = 14$ and where there is no time delay between the two retailers. Then retailer 1 could send two units right away to retailer 2 and therefore retailer 2 would be in a position where it could satisfy all possible demands including the maximum demand which we assumed here to be 8. But what if the time delay was still less than the one separating the retailers and the supplier but bigger than 0? We would have to introduce two new states to our model, $X_7$ which would represent the time delay when retailer 2 ships to retailer 1 and $X_8$ which would represent the opposite direction, see Fig. 4.1. This means adding two columns and two lines to our $A$ matrix and also updating the different equations (for instance $X_1$ would now also get goods from $X_7$ and not only $X_3$). This would also mean that we would have to introduce two more columns to our $B$ matrix, which would represent the orders. $u_3$ would represent the order $X_2$ receives from $X_1$ and $u_4$ would be the opposite order) and two more lines of course ($X_7$ and $X_8$). Therefore we have 8 states and 4 control variables now. Recall that:

$$x(t + 1) = Ax(t) + Bu(t) + Ed(t).$$

The matrices are as follows:

$$A = \begin{pmatrix}
1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix},
B = \begin{pmatrix}
0 & 0 & 0 & -1 \\
0 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix},
E = \begin{pmatrix}
-1 & 0 \\
0 & -1 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0
\end{pmatrix}.$$

Moreover the inventory cost of $X_7$ and $X_8$ are in our model the same as the one of $X_1$ and $X_2$ (i.e., 0.5).

**Remark 8.** Note that the inventory costs of $X_7$ and $X_8$ should at least be the ones of $X_1$ and $X_2$. Otherwise the shops would exchange their excess inventory in order to cut costs.

Are there other advantages? The most important question at this stage would be: besides enlarging the domain of feasible states, does the possibility of transshipping lower the expected cost for the states that were already feasible? In other terms, what would be the amount one would be willing to invest in order to have the option of transshipping?

In this model we obtain a solution with 11 regions ($|\mathcal{R}| = 11$) as opposed to 4 in the initial model. Like in the initial model the cases are symmetric, so we will only give the ones where $X_1$ orders from $X_2$.

The regions and their control laws are as follows: in all the regions the following constraints are satisfied:

$$8 \leq x_1 \leq 32,$$
\[ 8 \leq x_2 \leq 32, \]
\[ 0 \leq x_i \leq 8 \quad \forall i = 3, \ldots, 8, \]
\[ x_1 + x_8 \leq 32, \]
\[ x_2 + x_7 \leq 32. \]

The last two inequalities have been artificially introduced to the model, to avoid the fact that \(X_1\) and \(X_2\) store excess inventory in \(X_7\) and \(X_8\). This gave us more regions than expected.

**Region 1:**

\[
\sum_{i=1}^{8} x_i \leq 48,
\]
\[ x_1 + x_3 + x_4 + x_7 \geq 16, \]
\[ x_1 + x_2 \geq 16. \]

**Optimal control law** \(u\) **for Region 1:**

\[
u_1 = 24 - 0.5 \cdot \sum_{i=1}^{8} x_i,
\]
\[ u_2 = 24 - 0.5 \cdot \sum_{i=1}^{8} x_i,
\]
\[ u_3 = 0, \]
\[ u_4 = 0. \]

**Interpretation for Region 1:** Both retailers have less than 24 in the "pipeline", both order so that they have 24 in the pipeline.

**Region 2:**

\[ x_2 + x_5 + x_8 \leq 32, \]
\[ x_1 + x_3 + x_7 \leq 32, \]
\[ x_1 + x_3 + x_4 + x_7 \geq 16, \]
\[ 48 \leq \sum_{i=1}^{8} x_i \leq 56. \]
Optimal control law \( u \) for Region 2:

\[
\begin{align*}
  u_1 &= 0, \\
  u_2 &= 0, \\
  u_3 &= 0, \\
  u_4 &= 0.
\end{align*}
\]

**Interpretation for Region 2:** Both retailers have enough stock. They don’t order.

Region 3:

\[
\begin{align*}
  x_2 + x_8 &\leq 32, \\
  x_1 + x_3 + x_7 &\geq 8, \\
  x_1 + x_3 + x_7 + x_4 &\leq 16, \\
  32 &\leq \sum_{i=1}^{8} x_i \leq 48.
\end{align*}
\]

Optimal control law \( u \) for Region 3:

\[
\begin{align*}
  u_1 &= 24 - 0.5 \cdot \sum_{i=1}^{8} x_i, \\
  u_2 &= 24 - 0.5 \cdot \sum_{i=1}^{8} x_i, \\
  u_3 &= 16 - x_1 - x_2 - x_3 - x_7, \\
  u_4 &= 0.
\end{align*}
\]

**Interpretation for Region 3:** Retailer 1 won’t have enough stock to satisfy a demand of 8 in the next time step. Retailer 1 orders from Retailer 2 the inventory he is missing. Retailer 1 will have to order from the supplier in order to have the necessary inventory in two time steps and Retailer 2 as well given the fact that it had to provide goods for Retailer 1.

Region 4:

\[
\begin{align*}
  x_2 + x_8 &\leq 32, \\
  x_1 + x_3 + x_7 &\geq 8, \\
  \sum_{i\neq 6} x_i &\geq 48,
\end{align*}
\]
\[
\sum_{i=1}^{8} x_i \leq 56,
\]
\[
x_2 + x_5 + x_8 \geq 32.
\]

Optimal control law \(u\) for Region 4:
\[
\begin{align*}
    u_1 &= 0, \\
    u_2 &= 0, \\
    u_3 &= x_2 + x_5 + x_8 - 32, \\
    u_4 &= 0.
\end{align*}
\]

Interpretation for Region 4: Retailer 2 has too much inventory coming in. Retailer 1 has to relieve \(X_2\) of this burden by taking his excess inventory.

Region 5:
\[
\begin{align*}
    x_1 + x_3 + x_7 &\geq 8, \\
    \sum_{i \neq 6} x_i &\leq 48, \\
    \sum_{i=1}^{8} x_i &\geq 48, \\
    x_1 + x_3 + x_4 + x_7 &\leq 16.
\end{align*}
\]

Optimal control law \(u\) for Region 5:
\[
\begin{align*}
    u_1 &= 0, \\
    u_2 &= 0, \\
    u_3 &= 16 - x_1 - x_3 - x_4, \\
    u_4 &= 0.
\end{align*}
\]

Interpretation for Region 5: There’s enough inventory for the whole chain but more than enough in chain 2 and less than enough in chain 1. Retailer 1 orders the quantity he lacks from 2.

Region 6:
\[
\begin{align*}
    x_2 + x_8 &\leq 32, \\
    x_2 + x_5 + x_8 &\leq 32, \\
    56 &\leq \sum_{i=1}^{8} x_i \leq 64.
\end{align*}
\]

38
Optimal control law \( u \) for Region 6:

\[
\begin{align*}
 u_1 &= 0, \\
 u_2 &= 0, \\
 u_3 &= 32 - x_2 - x_5 - x_8, \\
 u_4 &= 0.
\end{align*}
\]

**Interpretation for Region 6:** There’s an excess of stock in chain 2. Retailer 1 helps out by ordering from 2.

Region 7:

\[
\begin{align*}
 x_2 + x_5 + x_8 &\leq 32, \\
 x_1 + x_3 + x_7 &\leq 32, \\
 56 &\leq \sum_{i=1}^{8} x_i \leq 64.
\end{align*}
\]

Optimal control law \( u \) for Region 7:

\[
\begin{align*}
 u_1 &= 0, \\
 u_2 &= 0, \\
 u_3 &= 0, \\
 u_4 &= 0.
\end{align*}
\]

**Interpretation for Region 7:** Similar to Region 2 there’s enough inventory in both chains. They don’t order.

**Conclusion**

Although there seem to be a lot of states, some of them are transient, i.e., they won’t be visited after a certain number of time steps (states which are expensive are transient). For example region 6, where there is excess inventory in one chain seems to be a region we would possibly encounter in a given initial state but not afterwards. The order-up-to policy (in this case order up to 24 like in the simple model) does not let us enter these regions. This policy seems to be the optimal policy in the transshipment model as well, the only difference here being the fact that we can have more initial feasible states. However by applying the above shown policies, we enter the non-transient (stable) states almost surely in a finite number of steps. Once this happens the optimal control policy becomes the order-up-to policy like in the simple model. This shows us that, in this example, transshipments would only be beneficial in extreme cases. This pretty much points to the direction that transshipments are not worthwhile in this case.
4.1. EVALUATING THE TRANSSHIPMENT OPTIONS

4.1 Evaluating the transshipment options

We will only be comparing the expected costs for the feasible states of the initial model because otherwise it doesn’t make sense to compare the two. To create a one-to-one map between the two we will fix $x_7$ and $x_8$ to 0. A logical assumption would be that the owner of such a chain would be willing to invest at least the minimal difference between the two costs. It is obvious to see that the cost of the model where the option of transshipments had been added is less than equal to the cost of the initial model. But does the difference equal 0 in some cases or is the difference always positive?

The first thing we had to do was to compare the expected cost-to-go of the model given an initial state and a fixed time horizon. Therefore in order to compute this cost given an initial state we propose the following algorithms:

First of all construct a function that would extract the cost matrix and the cost constant $(d_i)$ because they are separate in the data structure denoted below as "c". We did this by implementing the following MATLAB functions:

```matlab
function [matrix] = extract_vector(c)
    s=size(c.Ci);
    n=s(2);
    for k= 1:n
        matrix(k,:)= c.Ci{k};
    end;
    return;

    and

function [matrix] = extract_matrix(c)
    s=size(c.Bi);
    n=s(2);
    for k= 1:n
        matrix(k,:)= c.Bi{k};
    end;
    return;
```

Now that we have the cost matrix we can evaluate the objective value:

```matlab
function [cost] = objvalue(C,d,x,c) % C is the matrix
    u_star= mptctrl(c); %u_star(x) gives us the control variable vector
    % as a function of the state x if x is feasible and [] otherwise
```

40
4.1. **EVALUATING THE TRANSSHIPMENT OPTIONS**

```matlab
if (size(u_star(x)) == [0 0])
    if (size(x)==[6 1])
        cost = 100000;
    else cost=20000;
    end;
else
    J = C*x + d;
    cost=max(J);
end;
return;
```

Note that if the state is not feasible, `objvalue` returns a very large value but not the same for the initial model and for the transshipment model. This way if for the two models the state is not feasible for one or the other or both then their cost difference would still be high and wouldn’t interfere with the computation of the minimum cost difference between the two.

In the following we will use the fact that the minimal difference between two piecewise convex linear functions, say $f_1$ and $f_2$, is attained in the corresponding point of one of the vertices of $f_1$, provided that $f_1$ is always bigger or equal than $f_2$, see [5]. In our case this will happen in one of the vertices of the initial model’s cost function. Therefore we will evaluate the difference between the abscisse vertices of the initial model’s cost function and the corresponding points of the transshipment model’s cost function. We then take the minimum of all these values:

```matlab
function [cost] = vertex_diff(C_st,C_mig,d_st,d_mig,c_st,c_mig)
    P_st= c_st.Pfinal;
    P_mig= c_mig.Pfinal;
    V_st= extreme(P_st); %gives us a vector of the states whose cost is a vertex of the cost function

    n_st=size(V_st,1);

    for i= 1: n_st
        line=V_st(i,:);
        x=line';
        vec_st(i)=objvalue(C_st,d_st,x,c_st);
        vec_mig(i)=objvalue(C_mig,d_mig,[x;0;0],c_mig);
    end;
    cost=min(vec_st - vec_mig);
```

4.2 CONCLUSION

The result we get is unfortunately 0. This means that there are states where the expected cost-to-go \( J_N(\cdot) \) in the two models are identical and that risk averse investors would not be interested in investing in the option of transshipping. Alternatively, we could try to average the expected cost-to-go over a carefully selected initial states that we consider probable. For example, we can take \( x_1 \) and \( x_2 \) ranging from 16 and 32 and set all the other \( x_i \)'s to 0. These states seem like probable initial states (no incoming orders). In order to do this we wrote the following program:

```matlab
function vec = average_value(C_st,C_mig,d_st,d_mig,c_st,c_mig)
i=1;
for x1 = 16:32;
    for x2 = 16:32;
        x = [x1;x2;0;0;0;0];
        vec(i)=objvalue(C_st,d_st,x,c_st)-objvalue(C_mig,d_mig,[x;0;0],c_mig);
        i=i+1;
    end;
end;
return;

% st represents the simple model and mig the model with transshipments
```

The vector this program gives us back attains a maximum value of 0.09 and this is when \( x_1 = 32 \) and \( x_2 = 16 \) (which is small compared to the costs which are 24 and 24.09 respectively for 10 time steps). The vector also contains a lot of zeros. This leads us to believe that investing in transshipping is not worthwhile (the average gain is not that high). We will see later on a method to compute the benefit of transshipping in the long run using Markov chains.

4.2 Conclusion

This model has proven to be a useful feature if one would want to enlarge the domain of feasible states by sharing the inventories. Most of these states are transient and after a very small number of time steps the control policy will lead us to the recurrent states, i.e., the stable states. The policy will then become the same as for the simple model, i.e., the order-up-to policy will be applied. Transshipping does not cut the costs for these states and even for the transient states that are feasible for both models the difference in cost appears to be negligible. It is therefore our conclusion that investing in such an option is only worthwhile if one wants to extend the number of feasible states. This could prove to be useful if one day an unexpected demand appears in one of the retailers (more than the 8 which we said to be the maximum) and this way we would still be able to satisfy the demand given that the other retailer can cope with the emergency. However investing in the option of transshipping in order to cut costs does not prove to be worthwhile, at least in this simple case with independents demands and without the option of backlogs. In the next
4.2. CONCLUSION
Chapter we will do similar studies for more complicated cases, where the model combines the different extensions introduced until now.
Chapter 5

Integrating the models

We have, in the three previous Chapters, introduced some extensions in the initial model in order to be more consistent with reality. It is true that in reality there exists a certain correlation between the demands depending on the type of goods. Moreover the phenomena of backlogs and the possibility of transshipments between different actors of the supply chain are being widely considered by companies. The goal is to manipulate what we can in order to minimize the cost. In the case of transshipment, as we have seen in Chapter 4, we can be interested in looking for two main values; the minimum gain we obtain by investing in transshipment and, in this case the company owning the supply chain will obviously invest at least this amount of money in order to build up the transshipment. The other value is the average gain, it is the maximum the same company is willing to pay in order to make the extension, at least if it is not gambling on high risk. We already computed those numbers in Chapter 4. Here we are interested in doing the same but with more complicated models and by this we mean models that mix the extensions studied before.

5.1 Transshipments and location correlation

In this part we will mix the model constructed in Section 2.2 with the transshipment model. The goal is to have a model where the two retailers face location correlated demand without transshipments and a similar model where we introduce the option of transshipments between the two shops.

5.1.1 Characteristics of the model

Consider a supply chain with one supplier $W$ that delivers goods to two retailers $X_1$ and $X_2$ through two independent paths with a time delay of one. The ordering delay and the ordering cost to the supplier are zero for both retailers. The holding cost is set to one per unit of goods per unit of time in each shop and a discount factor of $\delta = 0.9$ is chosen. The time horizon is set to $N = 20$. Let us assume that the maximum order quantity is 8. The supply chain is represented by the graph of Fig. 5.1.
5.1. TRANSSHIPMENTS AND LOCATION CORRELATION

Figure 5.1: Model with location correlation without transshipment.

The matrices $A$, $B$ and $E$ of (1.3) are, with respect to what we just said,

$$
A = \begin{pmatrix}
1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\end{pmatrix},
B = \begin{pmatrix}
0 & 0 \\
0 & 0 \\
1 & 0 \\
0 & 1 \\
\end{pmatrix},
E = \begin{pmatrix}
-1 & 0 \\
0 & -1 \\
0 & 0 \\
0 & 0 \\
\end{pmatrix}.
$$

We also impose the following constraints on the states, where $x_i$ is the inventory of the node $X_i$:

$$
x_i \in [0; 24] \quad \text{for} \quad i = 1, 2, \quad x_i \in [0; 8] \quad \text{for} \quad i = 3, 4.
$$

In this model we consider a constant population of size 8. Each agent needs to have one unit of the good the company is producing on each time step so that the demand is negatively correlated among the two shops. Furthermore, with respect to what we introduced in Chapter 2 and for simplicity of the model we consider that the total demand can be distributed among shops 1 and 2 as follows:

<table>
<thead>
<tr>
<th>$d_1$</th>
<th>$d_2$</th>
<th>$p(d_1, d_2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>4</td>
<td>1/3</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>1/3</td>
</tr>
<tr>
<td>0</td>
<td>8</td>
<td>1/3</td>
</tr>
</tbody>
</table>

With this setup the feasible states for this problem are:

$$
8 \leq x_1 + x_3 \leq 24, \\
8 \leq x_2 + x_4 \leq 24, \\
x_i \geq 0, \quad i = 1, \ldots, 4,
$$

where the lower bound is here to prevent any unfilled demand and the upper bound prevents an excess of inventory in the shops. The order policy is in this case a simple order-up-to policy with hedging point $8 \cdot 2 = 16$. 45
5.1. TRANSSHIPMENTS AND LOCATION CORRELATION

Let us now introduce the transshipment option. \( X_1 \) and \( X_2 \) can now exchange part of their inventory or even the totality of it and this, without any time delays and ordering cost. One can think of a company that has two shops in the same neighborhood. The supply chain will then look like the graph in Fig. 5.2.

**Remark 9.** In order to avoid the fact that the two shops exchange inventories they do not have, we have to enforce that:

\[
\begin{align*}
  u_3 & \leq x_2, \\
  u_4 & \leq x_1.
\end{align*}
\]

![Figure 5.2: Model with location correlation and transshipments.](image)

The matrices of (1.3) are in this case:

\[
A = \begin{pmatrix}
1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix},
B = \begin{pmatrix}
0 & 0 & 1 & -1 \\
0 & 0 & -1 & 1 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0
\end{pmatrix},
E = \begin{pmatrix}
-1 & 0 \\
0 & -1 \\
0 & 0 \\
0 & 0
\end{pmatrix}.
\]

And the feasible states are now given by:

\[
16 \leq x_1 + x_2 + x_3 + x_4 \leq 48,
\]

\[
x_i \geq 0, \ i = 1, \ldots, 4.
\]

Here we can note the introduction of links between the retailers has a positive effect on the feasible region, and by that we mean that it makes it larger. To really understand why it is so, let us consider the following example.

**Example**

Suppose we are in the state \( \mathbf{x} = (4, 9, 2, 3)^T \). In the case without transshipments the shop \( X_1 \) is able to satisfy a maximum demand of \( 4 + 2 = 6 \) but as the maximum demand that \( X_1 \)
5.1. TRANSSHIPMENTS AND LOCATION CORRELATION

could face in our model is 8, this state is not feasible.
Now take the same state but in the case where there is the option of transshipment. The shop \( X_1 \) could satisfy a demand of \( 4 + 2 + u_3 - u_4 \) where \( u_3 \in [0; 9] \) is what he gets from \( X_2 \) and \( u_4 \in [0; 4] \) is what he gives to \( X_2 \). Using the same logic we have that \( X_2 \) is able to satisfy a demand of \( 9 + 3 + u_4 - u_3 \). By writing \( y = u_3 - u_4 \) one can really see that the two shops are exchanging part of their inventory.
To see whether this state is feasible or not we have to know if the two shops could be able to satisfy any possible sequences of random demand. By denoting \( d_1 \in \{0; 4; 8\} \) the demand that comes to the shop \( X_1 \) at time 1 then the demand that comes at shop \( X_2 \) at time 1 must be \( 8 - d_1 \) according to the negative correlation. Therefore we have to see if the following inequalities are satisfied:

\[
\begin{align*}
    d_1 &\leq 4 + 2 + y, \\
    (8 - d_1) &\leq 9 + 3 - y.
\end{align*}
\]

By adding the two lines we see that the demand could be fulfill and so the state is feasible.

5.1.2 Value of transshipments

We just have seen that introducing transshipment in the supply chain had a benefit aspect in that it makes the feasible region bigger in case backlogs are not allowed. We would like now to see whether it is valuable or not for the company.

By the approach introduced in Section 4.1, we see that a lower bound for the improvement of the expected cost-to-go caused by the transshipment option is 0.1 for the case where both holding costs are 1 and for a future of 20 time steps. It is small but nevertheless it is a positive number.

What about the average gain? A similar approximation using a number of representative initial states is obtained by the following algorithm:

```matlab
function [r] = locorr_average_value(C_st,C_mig,d_st,d_mig,c_st,c_mig)
i=1; r=0;
for x1 = 0:24;
    for x2 = 0:24;
        for x3 = 0:8;
            for x4 = 0:8;
                x = [x1;x2;x3;x4];
                y = x';
                if (x'*[1;0;1;0] >= 8)
                    if(x'*[0;1;0;1] >=8)
                        if(x'*[1;0;1;0] <= 15)
                            if(x'*[0;1;0;1] <= 15)
                                vec(i)=objvalue(C_st,d_st,x,c_st) - objvalue(C_mig,d_mig,x,c_mig);
                                i=i+1;
                            end;
                        end;
                    end;
                end;
            end;
        end;
    end;
end;
```
5.1. TRANSSHIPMENTS AND LOCATION CORRELATION

end;
end;
end;
end;
end;
end;
end;
end;
size= size(vec);
b= size(2)
n = vec*diag(ones(b));
r = n/b;
return;

This algorithm construct a vector that contains the difference of the value function (expected discounted cost-to-go) between the case without the option of transshipment and the case with this option, and this for every feasible state of the model without links between the shops. Then we take the average of these differences which gives us an approximation of the average gain we are looking for.

Remark 10. Note that since by introducing the option of transshipment we enlarge the possible ways of facing unknown demands, the cost-to-go of the model with the option of transshipment is at most the cost-to-go of the model without this option. Hence the vector of differences above has only positive component.

We found that the average gain with the setup we made is \( r = 0.7 \). Therefore a company will be willing to pay up to 0.7 unit of money in order to install the transshipments. Note that the maximum difference in expected cost can be easily computed by taking

\[ \text{max\_diff} = \text{max}(vec) \]

Here the maximum difference is 1.3.

Remark 11. In this case the transshipments are surely used in the time, but it can appear, as we saw before, that there exist models in which the state where we use the transshipments are transient and therefore there will be no minimum gain.

In the following graph we plotted the expected cost-to-go for 20 periods with a discount factor \( \delta = 0.9 \). The expected cost we plotted is a function of the inventory \( x_1 \) of shop 1 given that \( x_2 = 16 \), \( x_3 = 0 \) and \( x_4 = 0 \). First of all we plotted this for the model without transshipments and then for the model with such links. In this part we studied the case were there wasn’t any ordering cost for transshipments, now in the graph of Fig. 5.3 we slowly made them bigger and bigger to see what happens. Here is the legend of the graph:

- The blue line is the expected cost for the model without transshipments;
- The red line is the expected cost without any ordering cost on the transshipments;
5.1. TRANSSHIPMENTS AND LOCATION CORRELATION

- The magenta line is the expected cost with ordering cost set up to 0.5;
- Finally the dashed green line is the same as the previous but with an ordering cost of 1.

Figure 5.3: Discounted expected cost-to-go as a function of $x_1$ with different transshipment cost.

Conclusion

We can see that for our fixed value of $x_i, i = 1, \ldots, 3$ the discounted expected cost-to-go as a function of $x_1$ is uniformly smaller in the case where there is the option of transshipment compared to the case where there is not such an option. Furthermore, as the ordering cost for the transshipments increases, the expected cost-to-go of the former case will converge to the cost of the latter. This result is intuitive since, as the ordering cost increase, we will become less and less willing to use the option of transshipment.

Benefit per time step in the long run

We have computed the profit resulting from the introduction of the option of transshipments in the case of a finite time horizon. In the case of a time horizon that is large enough, we can use Markov chains in order to compute the expected profit per time step.

In some cases, a limit distribution $\Pi$ independent of the initial distribution $\Pi_0$ exists. If this limit distribution exists, then one only has to compute the inventory cost and order cost resulting from this limit distribution to get the limit overall cost per time step.

What we are willing to do here is to do such a study for the case without the option of transshipments and for the one where we do include transshipments and then compare the costs. In order to do so, we will need the following definitions and theorem:
5.1. TRANSSHIPMENTS AND LOCATION CORRELATION

**Definition 7.** A transition matrix $P$ of a Markov chain is called *regular*, if there exists some $n > 0$, such that $P^n > 0$.

**Definition 8.** A Markov chain is called *homogeneous*, if the transition probabilities are independent of the iteration. In other words, for all pairs $(i, j)$ and for all time horizon $n$, $P[X_n = j|X_{n-i} = i] = P[X_{n+k} = j|X_{n+k-1} = i]$ for all $k \geq 0$.

**Theorem 9.** Consider a homogeneous Markov chain and an initial distribution $\Pi_0$. If the transition probability matrix $P = [p_{ij}]$ is regular, then there exists a unique limit distribution $\Pi$ with $\lim_{n \to \infty} \Pi_n = \Pi$. $\bar{P} = \lim_{n \to \infty} P^n$ exists and every row of $\bar{P}$ equals $\Pi$.

First of all, consider the case **without the option of transshipments**. Because of the independence of the two paths, the optimal control law is a simple *order-up-to* policy with hedging point 16 in both shops. The recurrent states can easily be found and are the following:

\[
\begin{align*}
y_1 &= (0, 16, 8, 0), y_2 = (8, 8, 8, 0), y_3 = (4, 12, 8, 0), \\
y_4 &= (8, 8, 0, 8), y_5 = (16, 0, 0, 8), y_6 = (12, 4, 0, 8), \\
y_7 &= (4, 12, 4, 4), y_8 = (12, 4, 4, 4), y_9 = (8, 8, 4, 4).
\end{align*}
\]

The transition matrix is

\[
P_1 = \begin{pmatrix}
1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 \\
1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 \\
1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3
\end{pmatrix}
\]

$P_1$ is regular since $P_1^2 > 0$, in fact $P_1^2 = \bar{P} = \lim_{n \to \infty} P^n$ where

\[
\bar{P} = \begin{pmatrix}
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 \\
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 \\
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 \\
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 \\
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 \\
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 \\
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 \\
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 \\
1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9
\end{pmatrix}
\]

Here we can see that the stationary distribution is $\Pi = \left(\frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}\right)^T$. The state cost is in each case 16 and the order cost is 0 since there is no transshipment option. Therefore the overall expected cost per time step in the stationary distribution is

\[
\Pi \cdot (16, 16, 16, 16, 16, 16, 16, 16, 16)^T = 16.
\]
5.1. TRANSSHIPMENTS AND LOCATION CORRELATION

Now we will focus on the case with the option of transshipments. The optimal control law is,

- Order-up-to 24 jointly,
- \( u_3(x) = \max\{8 - x_1 - x_3, 0\} \),
- \( u_4(x) = \max\{8 - x_2 - x_4, 0\} \).

The recurrent states can easily be found and are: \( z_1 = (0, 8, 4, 4) \), \( z_2 = (8, 0, 4, 4) \) and \( z_3 = (4, 4, 4, 4) \). The probability transitions matrix \( P \) is

\[
P = \begin{pmatrix}
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3}
\end{pmatrix}
\]

We can directly see that the stationary distribution is \( \Gamma = \left( \frac{1}{3}, \frac{1}{3}, \frac{1}{3} \right)^T \).

Recall that the state cost is given by \( q \cdot x \) where \( q = (1, 1, 0, 0)^T \). Hence the state cost is 8 for the three recurrent states. Now let \( r \) denote a possible transshipment cost then the order cost is given by \( r \cdot x \) where \( r = (0, 0, r, r)^T \). The cost of each state is summarized in the following table:

<table>
<thead>
<tr>
<th>( x )</th>
<th>( u(x) )</th>
<th>state cost</th>
<th>order cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0,8,4,4)</td>
<td>(4,4,4,0)</td>
<td>8</td>
<td>4r</td>
</tr>
<tr>
<td>(8,0,4,4)</td>
<td>(4,4,0,4)</td>
<td>8</td>
<td>4r</td>
</tr>
<tr>
<td>(4,4,4,4)</td>
<td>(4,4,0,0)</td>
<td>8</td>
<td>0</td>
</tr>
</tbody>
</table>

In order to have the control cost per state in the stationary distribution, we multiply the control cost per state with the stationary distribution which gives us

\[
\Gamma \cdot (4r, 4r, 0) = \frac{8}{3} \cdot r.
\]

So the average cost per time step using the optimal control law is \( 8 + \frac{8}{3} \cdot r \).

Result

As a result we find that in the long run the option of transshipments brings profits whenever the following inequality holds:

\[
16 \geq 8 + \frac{8}{3} \cdot r
\]

which gives us \( r \leq 3 \).
The case we just studied was the case of a demand that was correlated with respect to location, now we want to come back to the model we studied in Section 2.1 and introduce in it links between the two different shops. Consider that there is no ordering cost and that the two shops can exchange part of their inventory without any time delay. The supply chain will therefore look like the one in Fig. 5.4:

![Supply chain diagram](image)

Figure 5.4: Supply chain of the model with time correlation and transshipments.

The matrices $A$ and $E$ of (1.3) are the same as the one of the model without transshipments but $B$ is now a $8 \times 4$ instead of a $8 \times 2$ matrix because there are two new order channels: shop 1 ordering at shop 2 and vice versa. The new $B$ matrix is:

$$
B = \begin{pmatrix}
0 & 0 & 1 & -1 \\
0 & 0 & -1 & 1 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}.
$$

We also impose the following constraints:

$$
x_i \in [0; 8] \text{ for } i = 1, 2,
$$

$$
x_i \in [0; 2] \text{ for } i = 3, 4, 5, 6 \text{ and }
$$

$$
x_i^h \in [0; 1] \text{ for } i = 1, 2.
$$
5.2. TIME CORRELATION AND TRANSSHIPMENT

The feasible region for this model is:

\[ 6 \leq x_1 + x_2 + x_3 + x_4 + x_5 + x_6 - x^h_1 - x^h_2 \leq 12. \]

By comparing this with the feasible region that we found in Section 2.1 we can, as before, see that the introduction of the links allow a larger set of feasible states. As an example one can take the state \( \mathbf{x} = (1, 2, 0, 0, 2, 1, 0)^T \) which is infeasible without transshipments because shop 1 would not be able to satisfy an hypothetic demand of 2 units of goods. If we introduce the links we see that this state fulfills the condition of feasibility. And in fact, we can see that as there is a lot inventory in the path of shop 2, this one could send a quantity of goods to shop 1.

Now that we conclude that introducing transshipment links had beneficial aspects on the feasible region we will see if in this model it is valuable to introduce them. By computing the program that gives us the minimum gain, we find that the minimum profit is zero. So there is no profit assured.

In order to approximate the average profit the following algorithm is implemented:

```matlab
function [r] = timecorr_average_value(C_st,C_mig,d_st,d_mig,c_st,c_mig)
    i=1; r=0;
    for x1 = 0:6;
        for x2 = 0:6;
            for x3 = 0:2;
                for x4 = 0:2;
                    for x5 = 0:2;
                        for x6 = 0:2;
                            for xh1 = 0:1;
                                for xh2 = 0:1;
                                    x = [x1;x2;x3;x4;x5;x6;xh1;xh2];
                                    y = x';
                                    if (x'*[1;0;1;0;0;-1;0] >= 3)
                                        if (x'*[1;0;1;0;0;1;0;-1] <= 6)
                                            if (x'*[0;1;0;0;1;1;0;-1] >= 3)
                                                if (x'*[0;1;0;0;1;1;0;-1] <= 6)
                                                    if (x'*[1;0;1;0;0;0;0;-1] >= 1)
                                                        if (x'*[0;1;0;0;1;0;0;-1] >= 1)
                                                            vec(i)=objvalue(C_st,d_st,x,c_st) - objvalue(C_mig,d_mig,x,c_mig);
                                                            i=i+1;
                                                        end;
                                                    end;
                                                end;
                                            end;
                                        end;
                                    end;
                                end;
                            end;
                        end;
                    end;
                end;
            end;
        end;
    end;
end;
```
5.2. TIME CORRELATION AND TRANSSSHIPMENT

We find that the average gain is \( r = 0.0023 \) which is very small, therefore it seems doubtful whether there is any gain from the transshipment option once the system has entered its recurrent states.

**Conclusion**

We saw in this chapter that there are models where introducing transshipments is significantly profitable whereas in others it is less so. Therefore we can say that the value of the transshipments depends entirely on the characteristics of the model we are considering.
Conclusions

We have extended the simple supply network model by introducing the following features: correlated demand, backlogs and transshipments. We derived the optimal policies in these and other cases via stochastic dynamic programming where we used the MPT toolbox [7] for solving recursively the resulting parametric Linear Programs.

We have also tried to incorporate two or more of these features together. This was seen in Chapter 3 where time correlation and transshipments as well as locational correlation and transshipments were mixed. We also attempted to analyse other mixtures, for instance backlogs and transshipments. However the solver in this case did not work or fed back incoherent results. We have experienced many difficulties with the solver which in the early stages of our work was not ready for such models. The main challenge of course was to successfully model and adapt our ideas within the solver’s framework, but in some cases this didn’t prove to be enough. Due to these difficulties we couldn’t always apply our theoretical knowledge in practice, the solver always being a limit to our imagination. However Dr. Marco Laumanns managed to solve most of these problems and helped us tremendously with the solver and many other things. We have therefore succeeded in determining optimal control policies and t.

One of the main challenges we faced was to discover if such features were financially worthwhile, i.e., was it beneficial for the owner of such a supply chain to invest in such projects such as transshipments. Backlogs and transshipments proved to be very effective in enlarging the feasible regions thus giving the owner more liberty in choosing inventory allocation. However we also saw that in a lot of cases, installing these features was not financially worthwhile. Simple transshipping didn’t prove to be beneficial but when we added the feature of locational correlation it did. Therefore deciding whether or not to invest in the feature of transshipping depends entirely on the structure of our supply chain.

The possible extensions we focused on are only a small fraction of what can and should be added to these models. Supply chaining proves to be a difficult research area. The model framework we worked on leads to problem formulations that are computationally very demanding due to the complexity to solve the corresponding parametric Linear Programs. Therefore developments in this area is often linked to algorithmic and numerical improvements.
Bibliography


