# Chapter 6

# Implementation

## 6.1 Introduction

After having described the design in detail in the previous chapter at the individual agent level, describing features, this chapter looks at the implementation details of the design. Under the implementation, various technologies, tools and methodologies are explained in detail.

## 6.2 Hardware for agents to operate

Either a single desktop computer or few standard desktop computers used to host the developed software in a distributed manner. If the software runs on few computers, those computers need to be interconnected through a Local Area Network.

## 6.3 Choice of the simulation framework and other software



After studying number of toolkits and frameworks available for agent oriented system modeling and development, MADKit (Multi Agent Development Kit) [13] was chosen to develop the proposed system. This is because, the toolkit being open source, development being active and specially, having a GUI based approach for agent management and monitoring. Also, one another fact is the toolkit being based on Java [21] where, the generated agent code can be easily edited to modify the agent functionality. Another reason to choose MADKit is, the lesser time required in implementing an already modeled agent system. The source code for agents also developed in Java programming language. The Open Forecast [16], which is an open source library with many statistical forecasting techniques built in, used to perform various forecasts by relevant agents.

## 6.4 Implementing individual Agents

Following the designs presented in the design chapter, individual agents are implemented.

## 6.5 Implementing the ontology

The global ontology and local ontology at each tour operator level designed as a class diagram as explained in the design chapter. Instances of the relevant classes serialized and stored as xml for later retrieval. These serialized objects represent updatable ontology since it is dumped and reused when agents are created again later.

## 6.6 Implementation of algorithms

There are few key algorithmic modules in the system. Thos start from the customer side when run in simulation mode. Other key algorithms reside in the supplier or the tour operator side. Each of those is explained in detail as follows.

### 6.6.1 Demand Simulation (Customer side)

When run in simulation mode, arbitrary demand need to be simulated which is known as perceived demand for the product sold by the supplier/tour operator. This demand is varied on daily basis as the auction or bidding progresses. There are three functions used in doing the simulation. They are as follows.



$$p = m*d + c \qquad (1)$$

Where, $p$ = price, $d$ = day of operation. Depending on the evaluation criteria, m and c are decided.

Even though formula in (1) represents a linear demand curve, in reality demand doesn't move in a linear fashion. Hence, to simulate the demand closer to a real situation is modeled by the following equation.

$$p_1 = k*(d-a)^2 + c_1 \qquad (2)$$

Above is a parabolic function and similar to equation (1), $p$ and $d$ represents the price and day of operation where all others are constants which are chosen during the evaluation.

It is quite natural that number of bids generated closely follows the demand or the price perceived for the products. In other words, when there is a high demand for a

product, usually there is higher number of requests coming from customers. Hence another two formulas are defined, similar to (1) and (2) with different coefficients used to generate the number of bids.

$$b = m_2 * d + c_2 \qquad (3)$$

$$b = k_2 * (d - a_1)^2 + c_3 \qquad (4)$$

In the above formulas, b is the number of bids generated where d is the current day in operation. Other coefficients are estimated and calculated during the simulation or evaluation.

### 6.6.2 Bargaining

During the simulation, when customer agents generate requests as per the previous section, it is in evident that they generate bids with much lower price than listed price. Hence, that comes to the supplier/tour operator side as a bargain. Hence, in the tour operator side, that request needs to be tackled and following is how that is done.

### 6.6.2.1 Algorithm

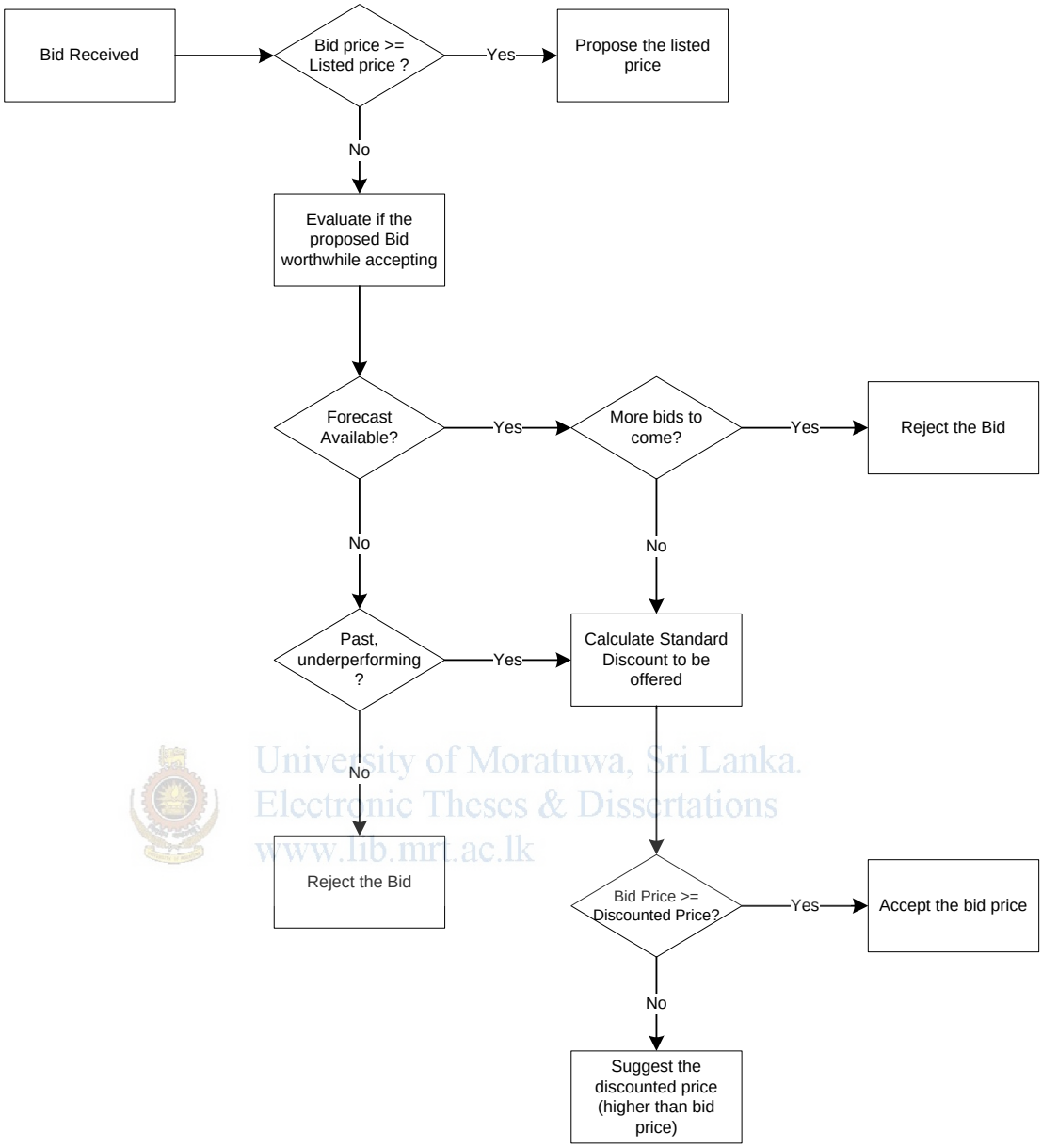




Figure 6.1: Bargain handling flow chart

Flow chart for the bargaining algorithm is depicted in Figure 6.1. In the figure, the "listed price" is the price which the tour operator sells a unit at the current day of operation which the bid is received.

### 6.6.3 Autonomous price adjustment (Supplier/Tour Operator)

With every sale being performed, the relevant agent performs the price analysis, and checks if how the price can be adjusted to generate more revenue. This particular check is done in daily basis, after closing operation for the day. In doing this, it is assumed the supply and demand curve for the product being sold to be one similar to the Figure 6.2.

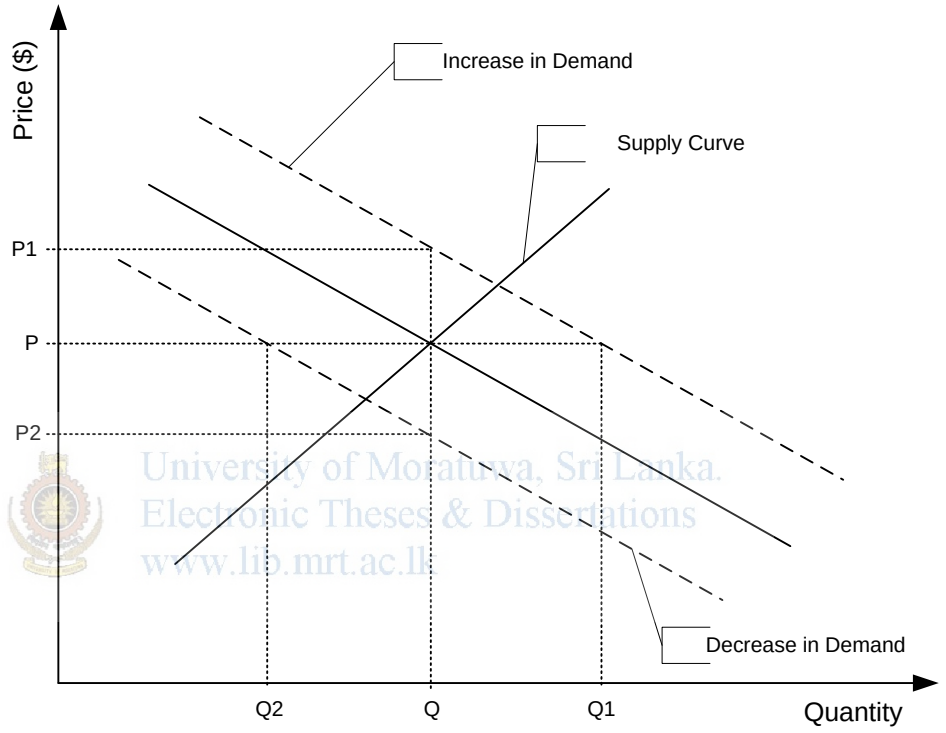



Figure 6.2: Demand Curve for Tour Operator/Supplier Items

In the demand curve depicted in Figure 6.2, when the business starts, supplier assumed to know Price P at which he can sell Q units maximum (Equilibrium point in supply and demand). Demand said to be increased, when customers are willing to pay Price P1, to obtain the same quantity Q. but in that case, if the supplier maintains his Price to be P, curve shows that he can sell Q1 units. This gives the chance to earn more revenue. Then supplier can evaluate which is profitable, either P1xQ or PxQ1. A similar methodology applies when demand decreases. Even within the loss, supplier still can choose either PxQ2 or P2xQ which generates higher revenue.

In all the demand calculations, it is assumed that supply remains fixed, but only the demand curve moves up and down.

During the simulation or evaluation, it is chosen, values for the linear demand curve, the slope, and other constants. Having such a formula, Figure 6.3 depicts how the price adjustment works in a flow chart.

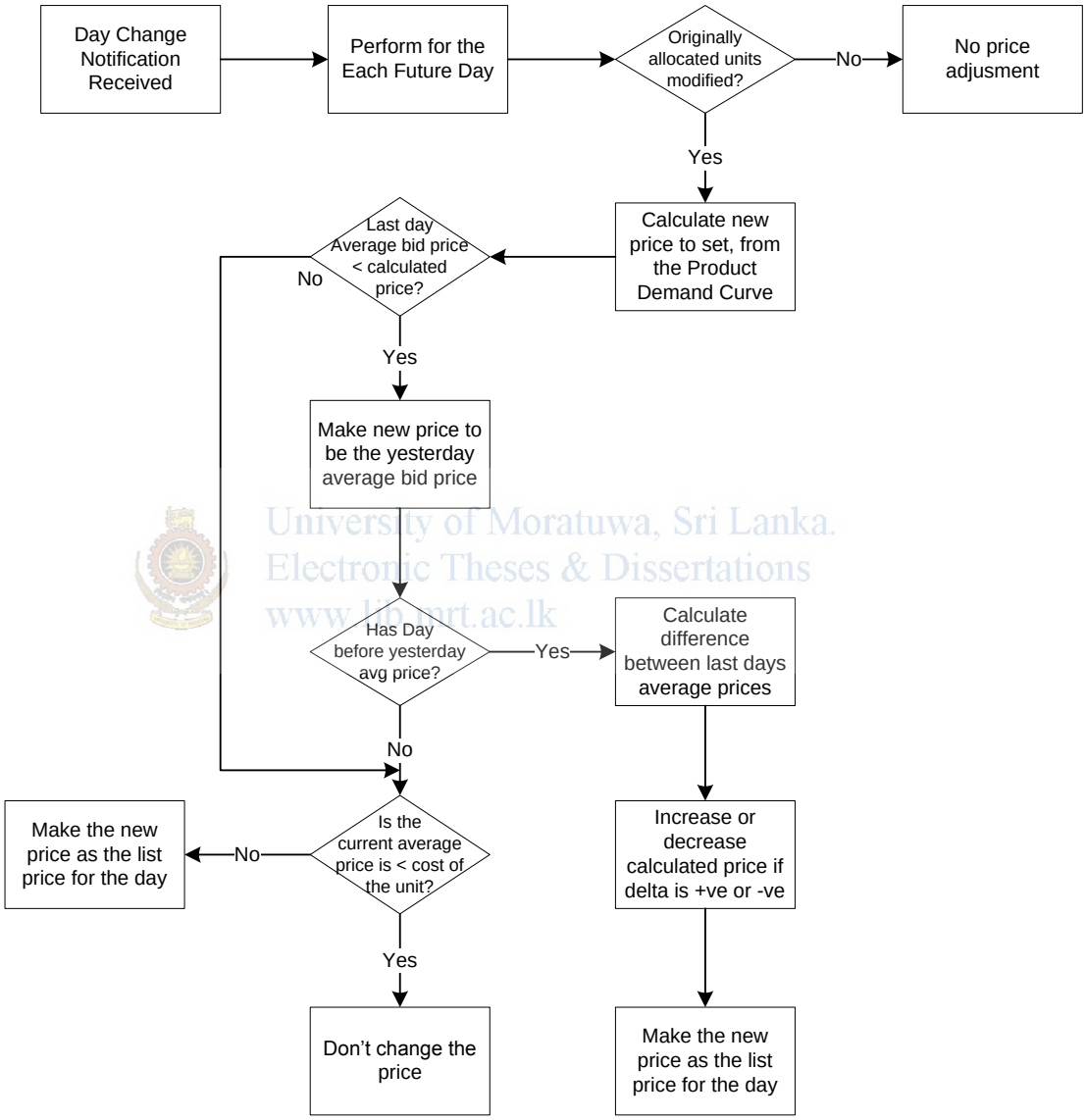




Figure 6.3: Price adjustment flow chart

### 6.6.4 Generating forecasts

As soon as a cycle of simulation or a business ends, relevant agents start performing forecast calculations. Since, the built model and simulation is of cyclic nature which

can be mapped to seasonality, and because there can also be a trend in future cycles, Triple Exponential Smoothing has been chosen as the function to generate the forecasts [8]. With that, Figure 6.4 shows the flow chart for forecasting number of bids that to come in future days and average bid price.

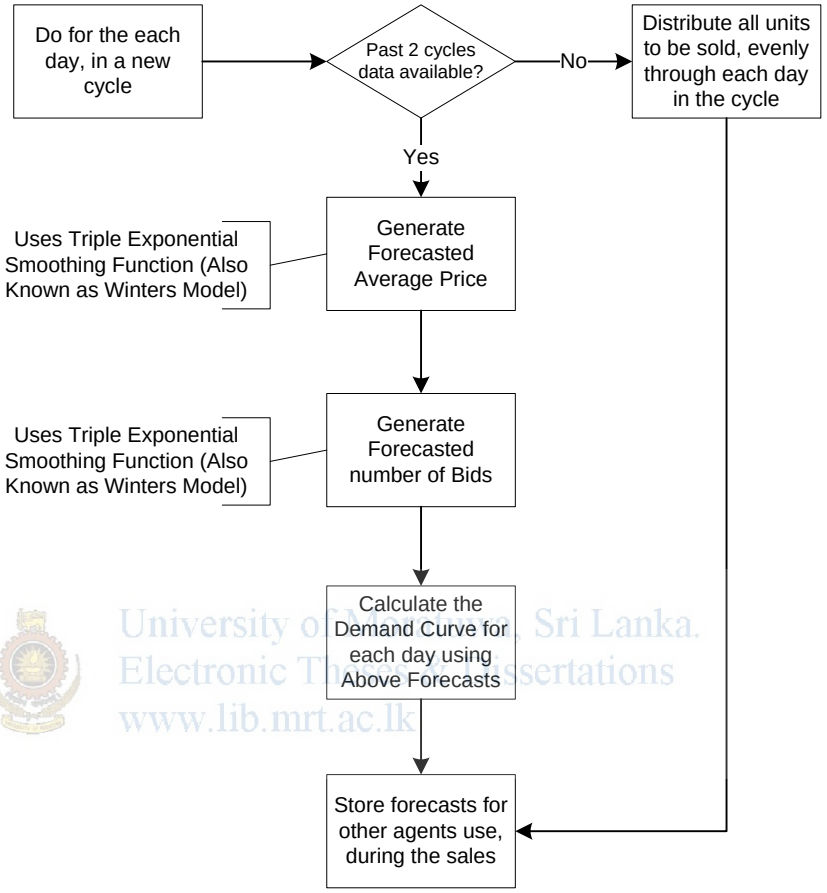




Figure 6.4: Flow chart for forecast calculation

## 6.7 Implementing the simulation software

Screens of the implemented software are presented in the Appendix B.

## 6.8 Summary

This chapter presented the details about the implementation of the proposed system. In this chapter, it was also highlighted, the toolkit or the framework to be used to implement the proposed model. MADKit has been chosen to make a faster implementation without worrying about the lower level communication protocols and agent monitoring system. Evaluation chapter comes next, with the details on evaluation strategy, setup …etc.