

A system dynamics model for the UK private house building supply chain

By Séverine Hong-Minh¹ and Jürgen Strohhecker²

¹ Wintershall AG, Friedrich-Ebert-Str. 160, 34119 Kassel, Germany
Tel: (49) 561 301 2496 Fax: (49) 561 301 1266,
E-mail: Severine.Hong-Minh@Wintershall.de
Formerly Logistics Systems Dynamics Group, Cardiff Business School.

² SIMCON GmbH, Friedrich-Ebert-Str. 70, 68723 Schwetzingen, Germany
E-mail:jstr@gmx.de

Abstract

This paper presents a system dynamics model for the private house building supply chains. The private house building industry in the UK is plagued by numerous problems, which affect its performance. A quick review of the main problems of the house building industry is given followed by a presentation of the similarities and differences of the house building industry with manufacturing in order to ensure that manufacturing solutions can be used for the house building industry.

A system dynamics model of a typical house building supply chain is developed using a set of sub-systems which can be “switched-off” to model specific scenarios. This model is based on an Inventory and Order Based Production and Control System (IOBPCS) which is widely used to represent manufacturing companies. The IOBPCS model is then further developed to incorporate specific features of house building supply chains. These features include an order and call-off mechanism, unfulfilled orders, mistrust and information enrichment. A review of tests to build up confidence in the model is then carried out and show that the model can be used with confidence in terms of its validity, credibility and generality.

Key words: System dynamics model, house building industry, supply chain

Introduction

The private house building industry in the UK is well-known for its problems. First of all, Sir Michael Latham highlighted in a report in 1994 entitled “Constructing the Team” the problems that plagues the construction industry in the UK and offered some guidelines to improve the performance of that industry (Latham, 1994). Few years later, the Egan report was published where good results of parts of the industry, for some specific construction projects, were acknowledged but it also reiterated the need for overall performance improvements (Egan, 1998). This report especially encouraged the construction industry to learn from other industries to improve performance.

This paper aims at presenting a generic system dynamics model that can be used to simulate different supply chain scenarios and assess the resulting performance. The model was developed during a research project involving companies representing the main players in a house building supply chain, i.e. a developer (named Home Builder), merchants and manufacturers. A case study approach was used for the data collection combining questionnaires, interviews and process mapping (Hong-Minh, 2001).

The first part of this paper will briefly present the house building industry problems and the need for change that the house building industry is facing in the UK. It will also present the similarities and differences the house building industry has with manufacturing. Based on these similarities, it can be assumed that some solutions for the manufacturing industry could improve the house building industry. Furthermore, it also allows the author to develop a generic model for the house building industry supply chains based on well-known production and distribution systems used in the manufacturing industry. This model will be presented in details followed by a brief overview of its possible use.

The UK private house building industry

Problem definition

The private house building industry suffers from many problems, some inherited from the construction industry in general such as cultural problems, others inherent to the speculative house building industry. This section does not intend to undertake an in depth study of all these problems, but to present a selection of problems to provide a clearer picture of the current state of the house building industry in the UK.

Customer needs

From a customer point of view, private house building may be perceived as not delivering the right product or the right quality or in the right location. Indeed, at present, new housing can be accused of being excessively standardised offering relatively low quality for expensive costs (Ball, 1996). This can be explained as house builders in the UK are focused on production and sales targets and not on their

customers' requirements (Roy and Gaze, 1999). A general lack of customisation of the products offered by house developers has been acknowledged (Barlow, 2000). Customers expect more specifications for their dwelling and to be offered more choices (Home Builder, 2000). Furthermore, regulations fail to stimulate the construction process and lengthen the lead-times (especially for planning application), although they do not hamper innovation as they are flexible enough to allow changes (Gann et al., 1998).

Construction stage

During the construction stage, supplier performance reveals itself as being a problem. Getting the right material to site at the right time for the right cost is not an easy task (Home Builder, 2000). Furthermore, the construction process is weather dependent, which can cause long delays. The private house building industry is also plagued like the rest of the construction industry by a skilled labour shortage (Latham, 1994; Ball, 1996). This shortage in skilled labour is directly linked to the lack of appropriate training (Gann and Senker, 1998).

Demand amplification

As for all industries, the house building industry suffers from demand amplification along its supply chain. Evidence of this phenomenon in the housing industry was reported by Lewis (1997) and is presented in Table 1. Lewis' (1997) example of demand amplification comes from the study of Ventair. Table 1 reveals that as demand is passed along the supply chain from customers to suppliers, it is distorted and amplified.

Average weekly demand from customers	Variability in weekly demand from customers	Average demand placed on suppliers and frequency	Variability in demands placed upon suppliers
48 per week	Up to 150%	400 every 6 weeks or 200 every 3 weeks	From 100% up to 300% depending upon frequency

Table 1 Amplification and variability in demand as it is passed along Ventair supply chain (Lewis, 1997)

The “mind-set”

There are other critiques to be made to house builders, those related to the house building culture and the way the companies involved in a housing project work with each other. Typically the relationships between the different disciplines involved in a construction project tend to be adversarial, as there is a lack of trust and commitment between parties (Bresnen, 1996; Holti, 1996; Larson, 1997; Barlow, 1998; Construction Productivity Network, 1998). The current “mind-set” may be defined as being a combination of poor communication, win-lose relationships and a lack of commitment (Hong-Minh et al., 2001).

Functional silo approach

Another generic problem to the house building industry lies in its functional silo approach to their work, which means that barriers can emerge between each discipline of their business. As a result such firms cannot establish a business process view of their work, neither can they fully focus on their customers' needs and requirements (Christopher, 1992). This problem is present not only at a company level but also at a project level involving several companies.

Better integration between all those involved in a construction project, such as customers, contractors and suppliers has been emphasised in the UK by Latham (Latham, 1994). The need for integration of processes within the UK construction industry supply chain process is also given prominence by the Office of Science and Technology (OST) (Office of Science and Technology, 1995).

Communication and learning

Due to the problem of adversarial relationships, communication between companies is very often not effective. In many cases, information is not always readily available (Latham, 1994) and is often incomplete or inconsistent (Construction Productivity Network, 1997). As stated by Latham (1994), there are serious deficiencies in information needed by builders. At the start of a project, requirements are often not fully defined, which impacts upon construction performance (Howell and Ballard, 1994; Bresnen, 1996).

The success or failure of a construction project's execution depends on the understanding of the information needs and requirements of the different parties (Love et al., 1999). Inadequate, incomplete and outdated information can lead to delays and extra costs during the design stage but also during the execution of a construction project.

The need for change

Until now, the private house building industry had very little incentive to change partly due to the fact that they make more profits on the land resale price more than on the product (houses) they sell. Furthermore, entry to market by more innovative foreign companies, i.e. Japanese, Dutch or Swedish, is made difficult by the lack of appropriate suppliers and labour (Barlow, 1999). They would therefore require to import innovative materials and trained labour which would considerably increase their costs. Thirdly, demand is higher than supply for new houses in the UK (Meikle, 1999).

This comfortable situation is however starting to change and the private house building industry will need to react. First of all, the environmental legislation will continue to require better performance from the dwellings. Currently these regulations are easily met or exceeded by the developers using traditional construction techniques. However as the regulations and standards will become tighter, developers will undoubtedly need to seek new ways of construction (Barlow, 1999). Secondly, the government's pressure to build on brownfield sites is increasing. A percentage as high as 45-50% of new housebuilding will have to be developed on brownfield sites has been quoted by the

government (Barlow, 1999). This has severe consequences on construction costs. Thirdly, the shortage of skilled labour is continuously increasing the construction costs (Barlow, 1999).

Therefore, ideas for improvement and change are required. As Gann (1998, p. 157) states, “*current levels of inefficiency and wasted materials, labour and time, as well as pollution, could be substantially lowered by streamlining supply chains and by introducing better management practices*”.

House building and manufacturing

A number of authors (e.g. Latham, 1994; Gann, 1996; Egan, 1998) argue that the house building industry could learn lessons from manufacturing industry such as the implementation of SCM principles. It is therefore important here to understand the specificity of the house building industry and its common features with the manufacturing industry.

The first specificity of the house building industry lies in the product itself (a house) as it is very large and immobile. The number of components needed to construct a house is four times greater than for assembling a car, depending on how the parts are counted, i.e. a house needs around 40,000 components, while a car only requires 10,000 components (Gann, 2000). The houses are built on a specific site with a mobile workforce which implies that the construction of houses can be highly influenced by weather conditions. Furthermore, in the UK, houses are expected to last long and are more likely to be regarded as an investment; people buy and sell houses relatively frequently (Gann, 1996; Ball, 1999; Ozaki, 1999). Contrary to manufactured products, which are transported to the market as finished goods, houses are assembled at the point of consumption.

The construction industry went through different stages: craft, machine age and digital age. Through these stages, the construction industry became industrialised. In other words it started using some manufacturing principles with the aim to raise the “*efficiency by rationalizing the process through the application of scientific method*” (Gann, 2000). However construction continued to suffer from high production costs induced by the fact that although the erection of structures and external work were more efficient, the finishing of internal work continued to be labour-intensive and performed in a traditional way (Gann, 2000).

Focusing on the processes for the construction of houses, Egan (1998) argues that the processes are repetitive. Repeat processes are comparable with the designing and planning of the production of a new car model, even more so for private house building as a specific house design will be built more than once. The house building industry may therefore be classified as a batching process with medium volumes using Hill’s (2000) classification. This implies that the house building is similar to the manufacturing industry in terms of processes with its specificity in terms of product.

Model description

In order to assess the impact of re-engineering scenarios on the house building performance, a generic model for house building supply chains has been developed. As the house building industry has similarities with the manufacturing industry, the model was based on a simple well-known model, an Inventory and Order Based Production Control System (IOBPCS) described later in this article, traditionally used to represent manufacturing company (Towill, 1982). The IOBPCS has been then further developed to incorporate the characteristics of the real system such as a site construction sub-system.

A system dynamics model was chosen as the aim of the model is to study the impact of re-engineering policies on demand amplification, i.e. a dynamic behaviour. Van Ackere et al. (1993) demonstrated that continuous simulation using system thinking is appropriate for studying the impact of BPR programmes as BPR has its origins in system engineering (Jenkins, 1971). Furthermore, Sterman (2000) states that models provide high quality outcome feedback. The role of simulation is to test the models (mental models) as without simulation, *“even the best conceptual models can only be tested and improved by relying on the learning feedback through the real world”* (Sterman, 2000, p. 37). However this real world feedback is often very long, costly, and the feedback can be inadequate and ambiguous. Therefore, simulation *“becomes the only reliable way to test hypotheses and evaluate the likely effects of policies”* (Sterman, 2000, p. 37). Therefore, system dynamics modelling and simulation can be accepted as an appropriate tool for the purpose of the research.

The model development process followed is shown in Figure 1. The model development process is iterative, starting from the problem description of the situation and the description of the aim of the model. The system development deals with the development of the model itself by identifying the structure and main components of the real system and translating them in terms of relationships. Finally the model is constantly validated during the development process.

A key step in model development is the problem description with the clear statement of the model's aim (Forrester, 1961). The aim of the model described in this paper is to assess the potential impact of different re-engineering policies on house building supply chains in terms of dynamic behaviour. Hence the aim of the model is to compare different scenarios derived from a main model and study in particular the effects these scenarios have on demand amplification.

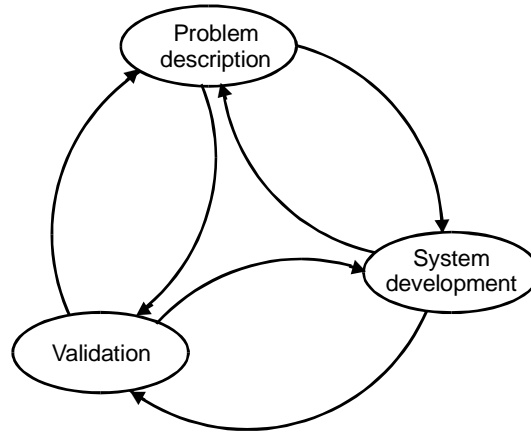


Figure 1 Model development process (Strohhecker, 1998)

Model structure

The structure overview of the model is presented in Figure 2 and is composed of six sub-systems. The demand represented by the construction plan is exogenous. This decision was made, as the aim of the model is to compare different scenarios in order to assess the impact of re-engineering policies on the dynamic behaviour by keeping the demand pattern identical. This demand is inputted into the Home Builder sub-system which represents the developer. Home Builder generates orders and call offs which are transmitted to the merchants. The merchants utilise this information to generate their own orders, which are sent to the manufacturer. The manufacturer then decides on the appropriate production level to fulfil these orders. A production allocation system has been implemented so as to decide which merchant should receive which quantities of material in the case of shortages. Then construction on site can take place using the material delivered by the merchants. Finally some performance indicators are calculated to assess the model.

The full documentation for the equations utilised in the model can be found in Appendix 1.

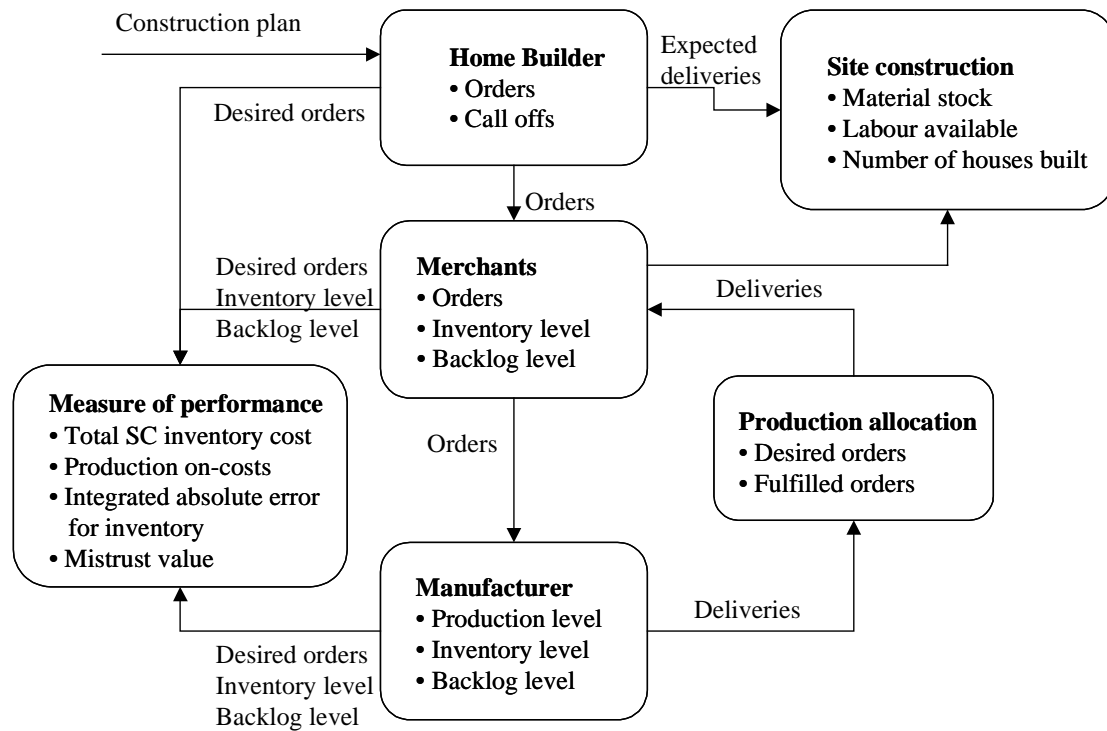


Figure 2 Model structure overview

Home Builder

Figure 3 represents the causal structure of the variables. This shows which variable influences what and how (same direction (+) or opposite direction (-)) (Sterman, 2000). Home Builder sub-system utilises the construction plan to generate the orders as shown in Figure 3. These orders after a pure time delay (which is represented by a crossed-line) become call offs. “Call off for the products” (C_p) is therefore equal to the orders (O) for houses delayed by λ_o multiplied by the product coefficient (p) (number of products needed to build one house).

$$C_{p(t)} = O_{(t-\lambda_o)} * p \tag{1}$$

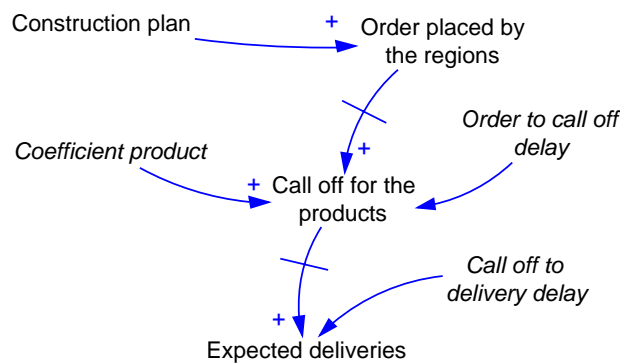


Figure 3 Causal structure for Home Builder

Finally expected deliveries is equal to “Call off for the products” delayed by a pure time delay of “Call off to delivery delay”.

Merchants and Manufacturer

Several merchants have been modelled, however all merchants follow the same structure and are exact duplicates. Just the quantities they order are different and based on “Coefficient merchant n”. Merchants and manufacturer sub-systems are similar and both based on Forrester’s production and distribution system (Forrester, 1958) and an Inventory and Order Based Production Control System (IOBPCS) model (Towill, 1982). The IOBPCS model, first analysed by Coyle (1977), has been extensively studied by the Logistics Systems Dynamics Group at Cardiff University since 1982 following a framework outlined by Naim and Towill (1993) (see Ferris and Towill, 1993; John et al., 1994; Cheema, 1994; Towill et al., 1997; Lewis, 1997; Disney et al., 2000).

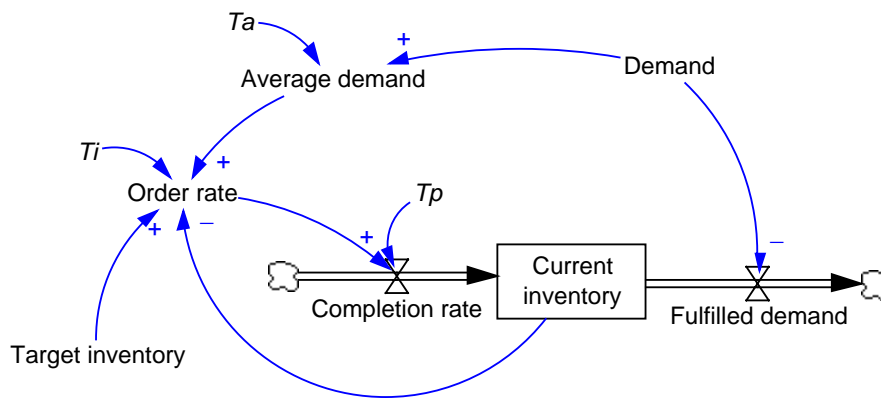


Figure 4 Stock and flow diagram for an IOBPCS model

Although the model is a simplified representation of the real world situation, it is held to be representative of much UK industrial practice associated with manual production control system (Coyle, 1977). Furthermore, the model includes variables that are common to a range of companies, such as production completion rate, stock levels and lead times (Berry, 1994) and it is able to replicate the dynamic behaviour of real world systems to a reasonable degree of accuracy (Edghill, 1990).

Figure 4 illustrates an IOBPCS model using a system dynamics stock and flow representation. The IOBPCS model represents an inventory and order based production system. This means that the level of production required (order rate) is based upon the level of demand which has been averaged over a period of time T_a and the level of current inventory in comparison with a target inventory. T_i represents the time to adjust the inventory and T_p the production delay.

The order rate is a function of the stock error and the average demand. Therefore, in terms of practical implementation issues, the assumption that current finished stock levels are accurately known has to be made (Edghill, 1990). The IOBPCS model is particularly suited to demands which are fairly constant (Lewis, 1997), however to cope

with higher variable demands, the target inventory level can be updated continually by making it a function of the average demand, which is the case in this work.

The IOBPCS model was used as a basis to represent merchants and manufacturer after discussions with several different managers from the companies modelled. They all declared that their ordering rule was solely based on the level of current demand and the level of stock in comparison with a target stock expressed as a function of the demand level. Only the distributor in the high-value fit-out supply chain model also declared to take into account the level of products in the pipeline.

However a feature present in Forrester's (1958) original model and not represented in the IOBPCS was added to the IOBPCS model. Usually an IOBPCS model is linear and it is assumed that whatever is asked for will come out of the pipeline after a delay. Forrester incorporated a backlog function utilising a non-linear representation. In this case if there are no products available in stock, then the products cannot be delivered. This feature takes into account the unfilled orders or order backlog (B_t), which are equal to the previous backlog (B_{t-1}) plus new orders (incoming call off) (C_I) minus shipment (processed call off) (P_B) as can be seen from Figures 5 and 6.

$$B_t = B_{t-1} + C_I - P_B \quad (2)$$

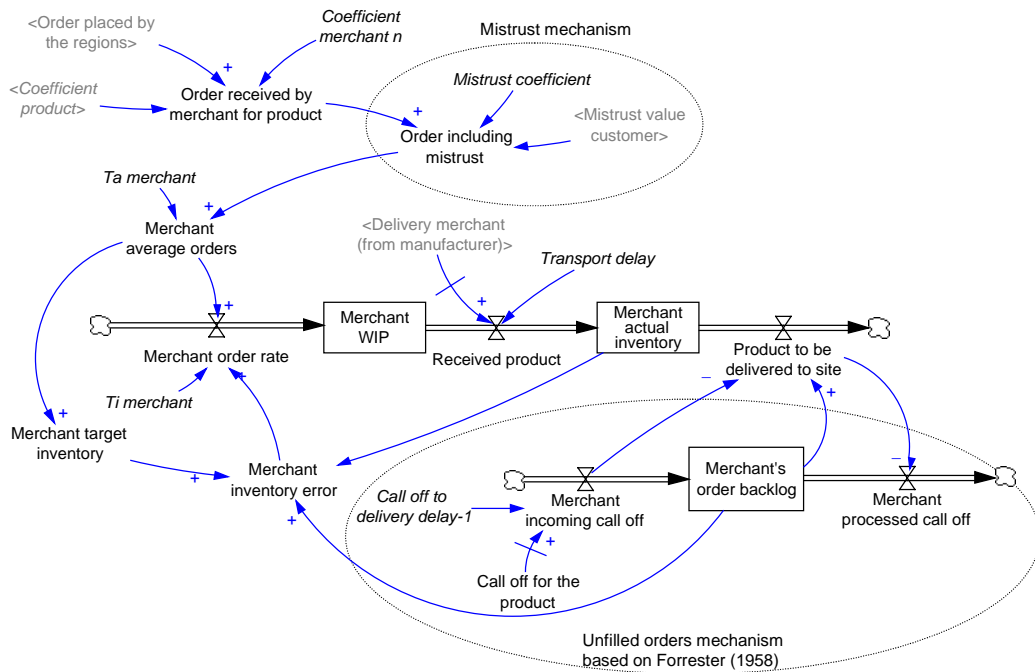


Figure 5 Simplified stock and flow diagram for Merchants

The equations for the ordering system are based on the IOBPCS and are as follows:

1. Average orders (A_O) equal the order including mistrust (O_m) smoothed over a period of time T_a :

$$A_{O(t)} = A_{O(t-1)} + \alpha A (O_{m(t)} - A_{O(t-1)}) \quad (3)$$

$$\text{where } \alpha A = \frac{1}{1 + \frac{T_a}{\Delta t}}$$

2. Order rate (R_O) is equal to the average orders (A_O) plus a fraction of any inventory errors (I_e):

$$R_{O(t)} = A_{O(t)} + \frac{I_{e(t-1)}}{T_i} \quad (4)$$

3. Completion rate (R_C) is equal to the order rate (R_O) divided by the production delay (T_p):

$$R_{C(t)} = \frac{R_O}{T_p} \quad (5)$$

4. Actual inventory (I_A) equals previous actual inventory plus the products produced (completion rate R_C) minus shipment (product to be delivered P_D):

$$I_{A(t)} = I_{A(t-1)} + R_{C(t)} - P_{D(t)} \quad (6)$$

Assuming $\Delta t = 1$

5. Inventory error (I_e) equals the target inventory (I_T) minus the actual inventory (I_A) plus the order backlog (B):

$$I_{e(t)} = I_T - I_{A(t)} + B_t \quad (7)$$

6. Target inventory (I_T) is a multiple of the average orders (A_O):

$$I_T = \beta A_O \quad (8)$$

The settings of T_a (time to average consumption), T_i (time to adjust inventory) and T_p (production delay) are based on John et al. (1994), Towill and Del Vecchio (1994), Mason-Jones et al. (1997) and Mason-Jones (1998). Several studies showed that in order to reduce demand amplification in IOBPCS models, a good setting would be $T_a = 2T_p$ and $T_i = T_p$ (John et al., 1994; Mason-Jones, 1998). However in Towill and Del Vecchio's (1994) work, it was seen that different settings could be used depending on the purpose of the model and where the company is positioned in the supply chain. Therefore the settings proposed by John et al. were used for the manufacturer and those of Towill and Del Vecchio for the merchants, i.e. $T_a = T_i = 2T_p$.

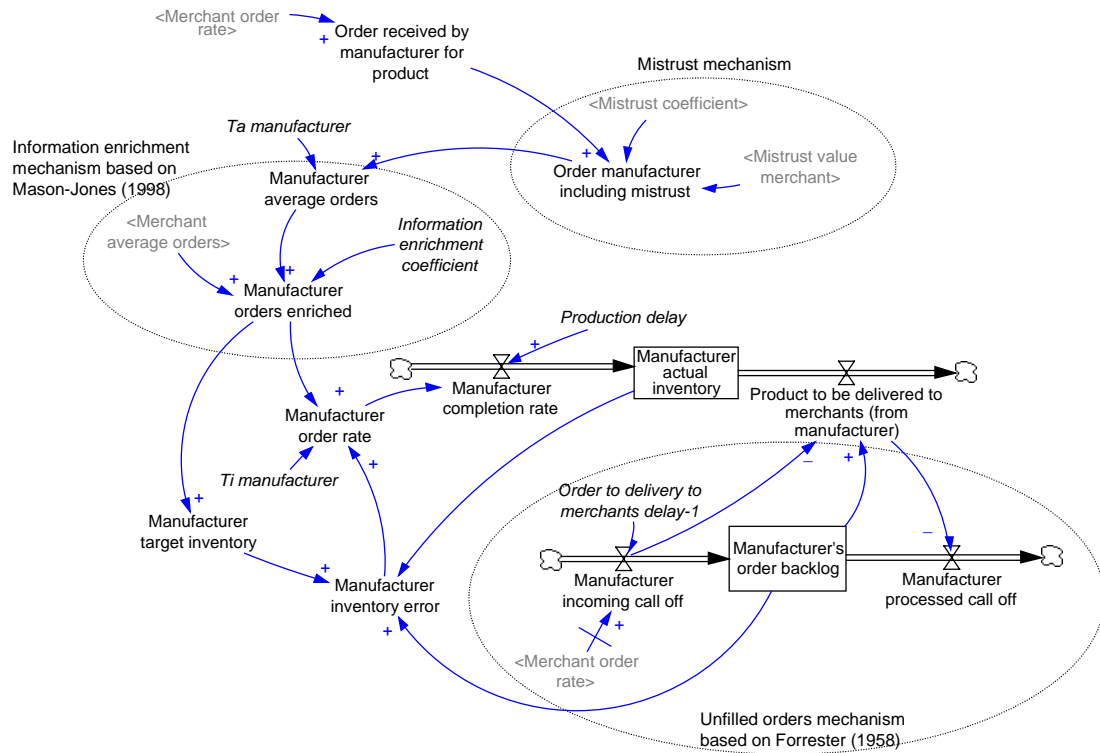


Figure 6 Simplified stock and flow diagram for Manufacturers

A **mistrust mechanism** has been included in the model as can be seen from Figures 5 and 6. Mistrust has been included to the model, to reflect the particularity of adversarial relationships in the house building industry. The modelling of mistrust is based on real life observations during the research work involving case studies and on Sterman (2000). Mistrust is understood as being the lack of trust between trading partners. Very often this lack of trust is especially tangible when customers do not receive the full quantity of what they have ordered. Instead of trusting the supplier that he will deliver the missing products as soon as they become available, customers over-order to make sure that they will receive the real quantities they need. This principle has therefore been reflected in the model as follows: whenever the customers do not receive the full delivery of what they have ordered (O), the next order (O_m) they will place will be increased by a percentage (mistrust coefficient κ_m) of the quantity of product undelivered (mistrust value M). κ_m equals 1 when mistrust is set at 100% and 0 in the case of no mistrust. The calculation of the “mistrust value” is explained later in this paper.

$$O_m = O * (1 + (\kappa_m M)) \quad (9)$$

An **information enrichment** mechanism has also been incorporated into the model and can, as easily as in the case of mistrust, be switched off. The information enrichment mechanism allows to simulate scenarios with improved communication between the supply chain partners. This mechanism, based on Mason-Jones’s (1998) work, has been placed at the manufacturer level and allows it to utilise the smoothed market demand (merchant average orders A_{Od}) and the smoothed customer orders (manufacturer average orders A_{Om}) to decide on a desired level of demand (manufacturer orders

enriched O_E). The information enrichment coefficient (δ) in Figure 6 determines how much of the market demand needs to be used.

$$O_E = \delta A_{Od} + ((1 - \delta) * A_{Om}) \quad (10)$$

Therefore when $\delta = 1$ (i.e. 100% information enrichment), the manufacturer is relying only on the market sales data from Home Builder to decide how many products to produce and could thus easily respond incorrectly to customer demand (merchants). If $\delta = 0.5$ (i.e. 50% information enrichment) then the manufacturer will base his decision on 50% of the market demand and 50% of its customer demand. Finally setting $\delta = 0$ switches off the information enrichment mechanism and the manufacturer relies solely on its own customer demand (merchants).

Production allocation

A production allocation sub-system has been developed in order to replicate the allocation of products from one manufacturer to several customers (merchants). Production allocation sub-system takes into account the overall demand (total merchant order rate) from the different merchants simulated. The rule followed for the production allocation is based on real life observations made during the case studies, where, whenever there is insufficient product available to satisfy the demands of all the merchants, the distribution of the available product will be carried out based on a priority system. Here, the priority system has been set up so as to serve the largest customers first, then the second largest and so on until there is no more product available. The customer ordering the greatest quantities is defined as the largest customer.

The priority is modelled using the “coefficient merchant” which is used to distribute Home Builder overall demand among the different merchants. Therefore if “coefficient merchant 3” is equal to 0.6, it means that Merchant 3 will receive 60% of Home Builder orders and therefore represent 60% of the orders passed on to the manufacturer.

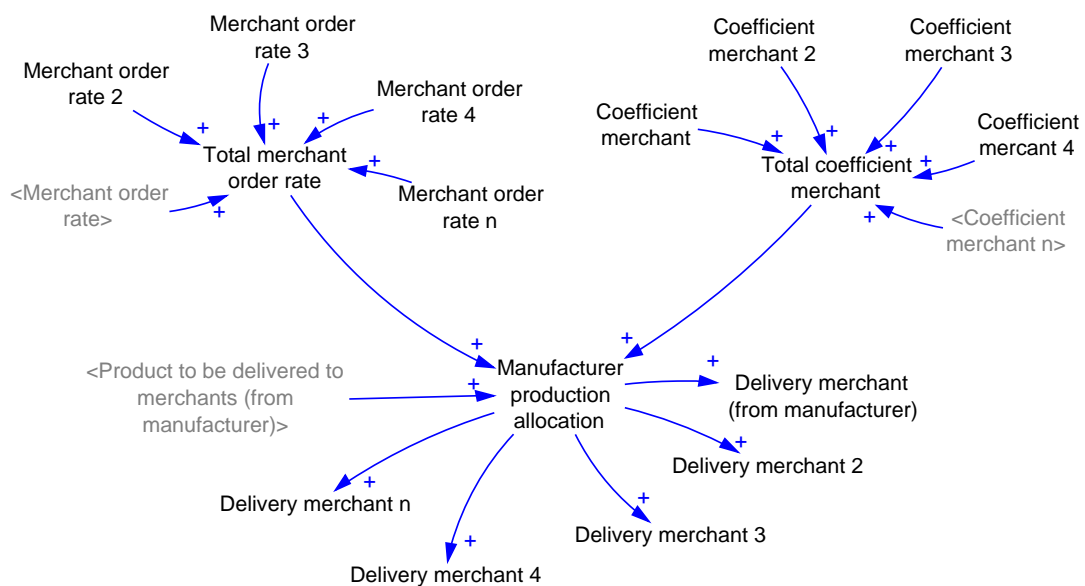


Figure 7 Simplified causal loop for production allocation

As can be seen from Figure 7, production allocation utilises the “total merchant order rate” as the overall demand, the “total coefficient merchant” to prioritise the allocation and finally “product to be delivered to merchants” as the quantity of products available for allocation. The results of the production allocation are then given in “delivery to merchant n” variables. Production allocation was calculated using ALLOCATE BY PRIORITY function available in Vensim® (see Appendix 1).

Site construction

The site construction sub-system is principally made of two components; labour and material. The demand for *labour* (L_D) is derived from the call off for the *products* (demand) (C_p). C_p is transformed into a demand for labour using both product (p) and labour coefficients (l) as can be seen from Figure 8 and Equation (11). For example, if the call for *products* equals 200 with a product coefficient of 2, it means that there is a demand for *house* equal to 100. Now if 3 labours are needed to build one house, then the demand for labour will be equal to 300.

$$L_D = \left(\frac{C_p}{p} \right) * l \quad (11)$$

A backlog system similar to the unfilled orders mechanism is used to register unfulfilled demand in labour. The houses can only be constructed if there are enough labour and enough products available; in case of discrepancies, the lowest of the two variables (labour and product) dictates how many houses can be built. The remaining labour or products will then stay idle on site until more products or labour become available to finish the work. After a “construction time delay” (represented by a pure time delay) the houses become weatherproofed houses and the labour is then freed and re-injected as free labour. Therefore, it is assumed that once the labour starts working on site, it will remain on site until the houses are built, only then will it be become labour available for use.

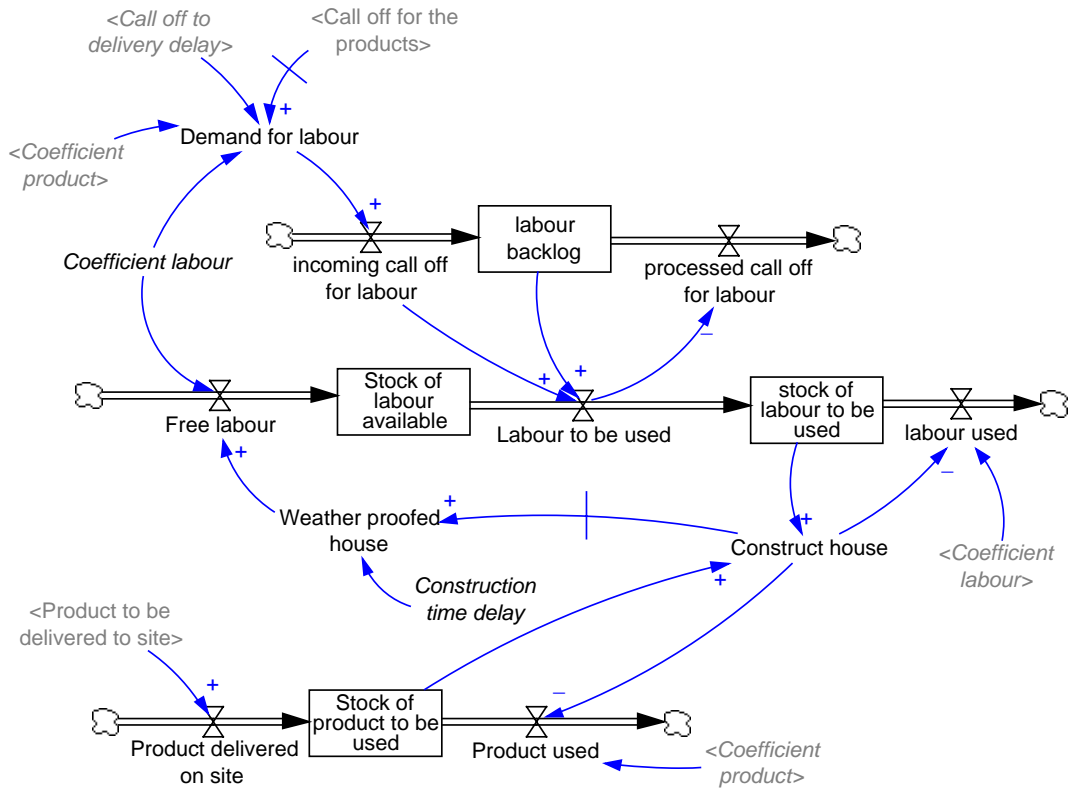


Figure 8 Simplified stock and flow diagram for site construction

In the case of a supply-and-fit system, the stock of product to be used on site will first be checked before calling off any labour. Therefore if there are fewer products available than the actual call off for labour, only the labour necessary for the assembly of the available products will be called off. In this case labour is only called off when there is really work to be done whereas in the previous case (labour only), labour could be called off and stay idle on site until products to be assembled became available.

Measure of performance

Several measures of performance are calculated in the model so as to assess each scenario. The performance criteria calculated directly in the model are the total inventory cost, the production on-costs, the integrated absolute error for inventory and the mistrust value.

The total supply chain inventory cost, presented in Figure 9, encompasses the inventory costs for the merchants and for the manufacturer. Therefore, the sub-system utilised for the merchant is used for each merchant modelled. The calculation of the inventory costs is based on the assumption that the cost for holding stock is different from the cost of being out of stock. The inventory costs (I_C) is therefore a function of the cost in stock (a), the level of stock (actual inventory I_A), the cost of stock out (b) and the level of backlog (B):

$$I_C = aI_A + bB \quad (12)$$

The inventory cost for each organisation is integrated and then summed to calculate the total supply chain inventory cost.

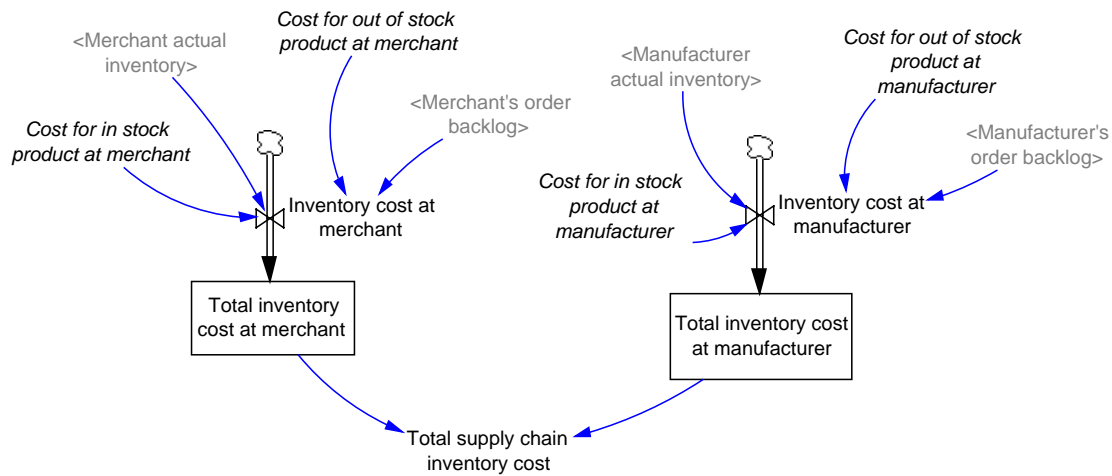


Figure 9 Simplified stock and flow diagram for total supply chain inventory cost

The next performance criteria are the production on-costs and the integrated absolute error (IAE) for inventory, which are calculated in a similar fashion. Both criteria accumulate the absolute error between a given behaviour and the actual behaviour. Production on-costs is concerned with the order rate at the manufacturer level. The manufacturer order rate is compared with the actual end-customer orders (the regions) and after having integrated the absolute error, the result is cubed so as to give the production on-costs (Berry, 1994). It can be seen from Figure 10 that the manufacturer order rate (for products) needs first to be converted into order rate for houses using the product coefficient.

The IAE for inventory simply uses the inventory error calculated using Equation (7) and cumulates the absolute error.

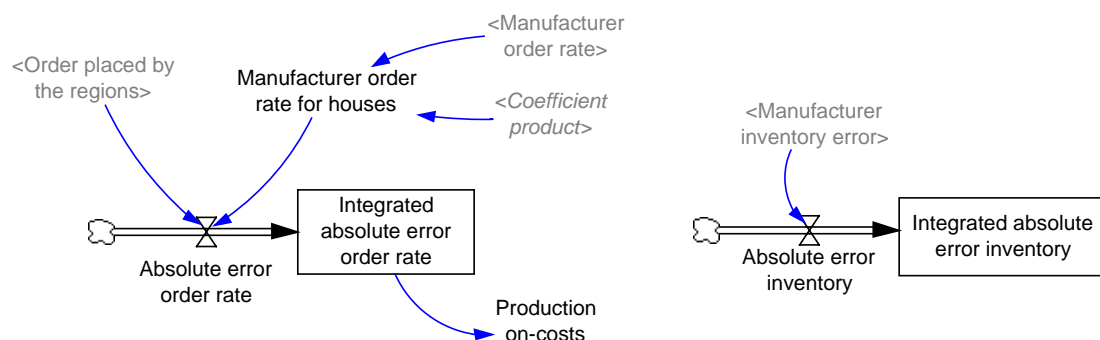


Figure 10 Simplified stock and flow diagram for production on-costs and integrated absolute error for inventory

Finally, the mistrust values are also calculated. There are two mistrust values, one for Home Builder at the site (customer) and one for the merchants as seen from Figure 11. The first one represents the percentage of products that Home Builder did not receive

from the merchants while the second one considers the percentage of missing products that the merchants should have received from the manufacturer.

The expected deliveries to site are calculated from the order received by the merchant and delayed by a pure time delay (call off to delivery delay). The quantity of missing product can then be calculated by comparing the quantity of products actually delivered on site to what was expected. The percentage of the product missing can then be calculated.

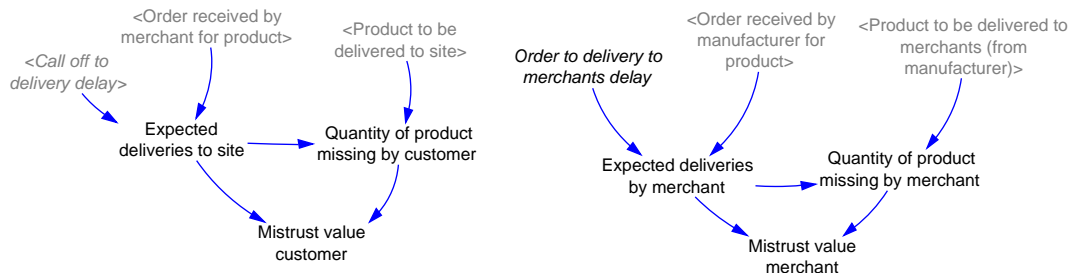


Figure 11 Simplified causal structure for mistrust value

Validation and verification

Model validation and verification are very important issues for all modellers. However, Sterman (2000) clearly states that, “no model can ever be verified or validated”, as by definition they are a simplified representation of the reality and therefore vary from the real world in many different ways. Forrester and Senge (1980) adds that, “validation is the process of establishing confidence in the soundness and usefulness of a model. Validation begins as the model builder accumulates confidence that a model behaves plausibly and generates problem symptoms or modes of behavior seen in the system.” Furthermore a model can be considered as realistic “to the extent that it can be adequately interpreted, understood, and accepted by other points of view” (Churchman, 1973).

Solberg (1992), who promotes simple models, declares that, “the power of a model or modelling technique is a function of validity, credibility, and generality. Usually the simplest model which expresses a valid relation will be the most powerful”. Hence, models need, one way or another, to be validated, however there is no single test which would allow the modellers to assert that their models have been validated. Rather, the level of confidence in the model can increase gradually as the model passes more tests (Forrester and Senge, 1980).

A wide range of tests to build confidence in the model have been developed (e.g. Forrester and Senge, 1980; Barlas, 1989; Barlas, 1990; Barlas, 1996), of which Sterman (2000) presents a summary. These twelve tests are as follows:

The boundary adequacy test is concerned with the appropriateness of the model’s boundary. For this test the main question to answer is if the appropriate concepts have been included in the model to address the problem. As stated previously, the aim of the model is to compare different supply chain scenarios where a few parameters change,

and assess the impact of these changes on the dynamic behaviour. The area concerned is the house building supply chain. The main players in the supply chain have been represented (i.e. Home Builder, merchants and distributor). The focus of the model is placed upon the material and information flow. Both flows have been repeatedly and successfully studied previously using a member of the IOBPCS model, also used here. Furthermore, although simplified, the model was considered as representative of the real situation by managers from the companies modelled.

The structure assessment test, as its name indicates, is concerned with the consistency of the structure of the model by verifying if the structure of the model represents the real system. As previously presented, the basis of the model uses two IOBPCS models to represent merchants and manufacturer. In addition, Coyle (1977) and Edghill (1990) asserted that the IOBPCS model was representative of the UK manufacturing practice and replicate the dynamic behaviour of real world systems to a reasonable degree of accuracy. Finally some changes have been made to the basic IOBPCS model to take into account some real life issues such as unfulfilled orders and the distinction between orders and call offs.

The dimensional consistency test examines if the units of measures used in the model are consistent. This was carried out using the dimensional consistency function available in Vensim®.

The parameter assessment test compares the model parameters to knowledge of the real system to determine if the parameters correspond conceptually and numerically to real life. The parameter values have been set for each scenario and principally based on real data collected during the case study research.

The extreme condition test analyses the behaviour of the model under extreme conditions to verify that the model behaves in a realistic fashion. In the model studied, the extreme condition test was carried out for an extremely high demand, the stock level and the amount of labour available then dropped to zero. The number of houses completed reflected the labour capacity. However, as no manufacturing capacity restriction has been made in the model, the stock level still recovered using an extremely high production level. Manufacturing capacity restriction was not introduced into the model so as to keep it as simple as possible, and also because the purpose of the model is not to study capacity issues but broader issues of dynamic behaviour.

The integration error test verifies if the time step utilised for the simulation and the integration method are appropriate for the purpose of the model. In the present case the time step was set at one day, however in order to test the model, the time step has been cut in half and the results compared. For the purpose of this model, the differences were marginal. The test was also carried out using a different integration method. Furthermore, as the purpose of the model is to compare different scenarios but keeping the time step and the integration method constant, the choice for the time step and the integration method become secondary.

The behaviour reproduction test assesses the model's ability to reproduce the behaviour of the real system. Although, the behaviour reproduction test is generally used for a model whose purpose is to reproduce very accurately the real world system

by comparing simulation results and real historical data. This test can also be used to assess if the model is behaving as expected. Therefore, the model was presented to the companies simulated and they all agreed that it represented their supply chain.

As can be seen from Figure 12, the model is creating a demand amplification pattern. The pure time delay between the orders and the calling off can be clearly seen. The implications of using an order and call off system can be seen at the merchants level with an increased order rate, which is based on “order for products” from the regional office followed by a drop. A delay at the manufacturer level can also be observed as well as amplification in the demand pattern.

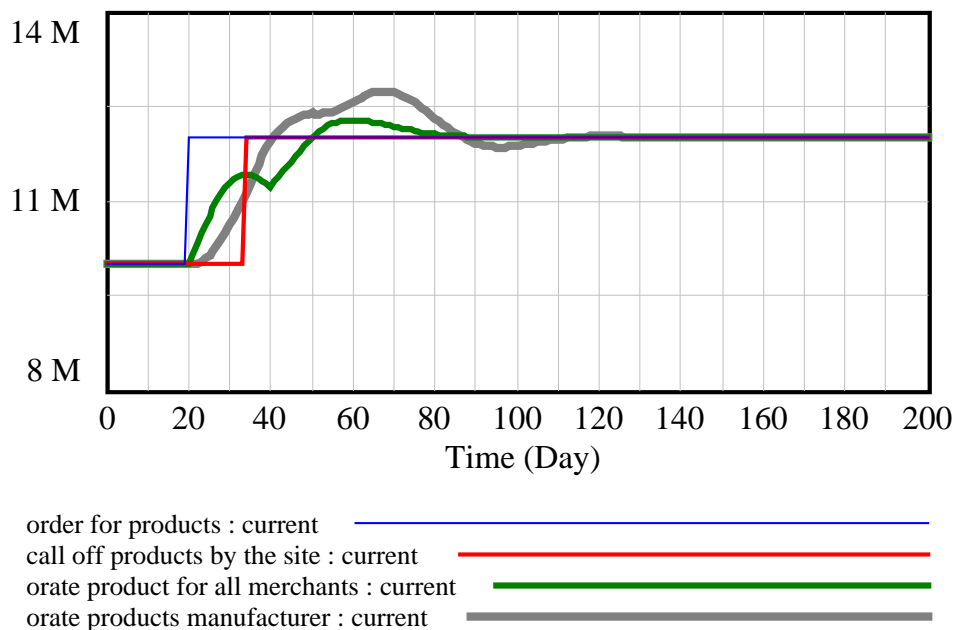


Figure 12 Sample results comparing demand, call-off and order rates for merchants and manufacturer

The behaviour anomaly test, which examines the importance of specific relationships by deleting or modifying them, was utilised during the model development process. This test helped in analysing the influence of specific variables.

The family-member test asks whether a model could be used to represent other more particular models. As stated previously, the IOBPCS used in the model is already part of a model family. Furthermore, the model was used to simulate different scenarios by “switching on or off” the relevant sub-systems and therefore the model proves to be a general one, that can be adapted to represent specific members.

The surprise behaviour test is concerned with unexpected behaviour displayed by the model. The test is passed when the behaviour does indeed occur in the real world. This was the case for the build up of stock observed at the merchants and manufacturer level. When using a normal IOBPCS model, the stock first diminishes before regaining a stable state. However in the model studied here, the stock first increases and then

diminishes. This is, however, happening in the real world where companies stock up in advance of a large order.

The sensitivity analysis tests the robustness of the model. As the model is based on an IOBPCS model, sensitivity analysis have already been carried out and showed the robustness of the IOBPCS model (e.g. Edghill, 1990; Disney et al., 2000). However, a new sensitivity analysis has also been carried out for the overall model. The analysis took into consideration the three parameters influencing the ordering policy, i.e. T_a (time to average consumption), T_i (time to adjust inventory) and production delay. The analysis was carried out to study the impact of these parameters on the manufacturer order rate and the stock of products at the manufacturer level. T_a was tested for a range from 1 to 60, T_i from 2 to 30 and production delay from 1 to 30, which is the maximum range utilised during the simulations.

Figures 13 and 14 present the results of the sensitivity analysis for the production delay. The other sensitivity analysis graphs for T_i and T_a can be found in Appendix 2. The sensitivity analysis showed that in all cases, a stable state is reached and that it is not sensitive to changes in parameters values. Therefore the model can be considered as being robust.

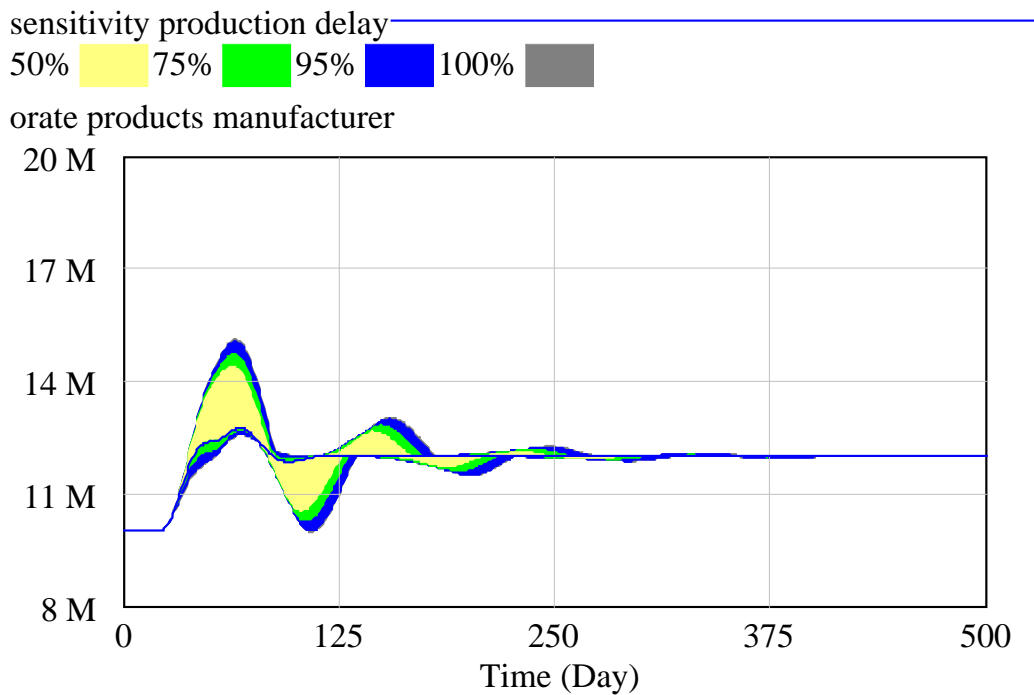


Figure 13 Sensitivity analysis for production delay (1 to 30) on the manufacturer's inventory level

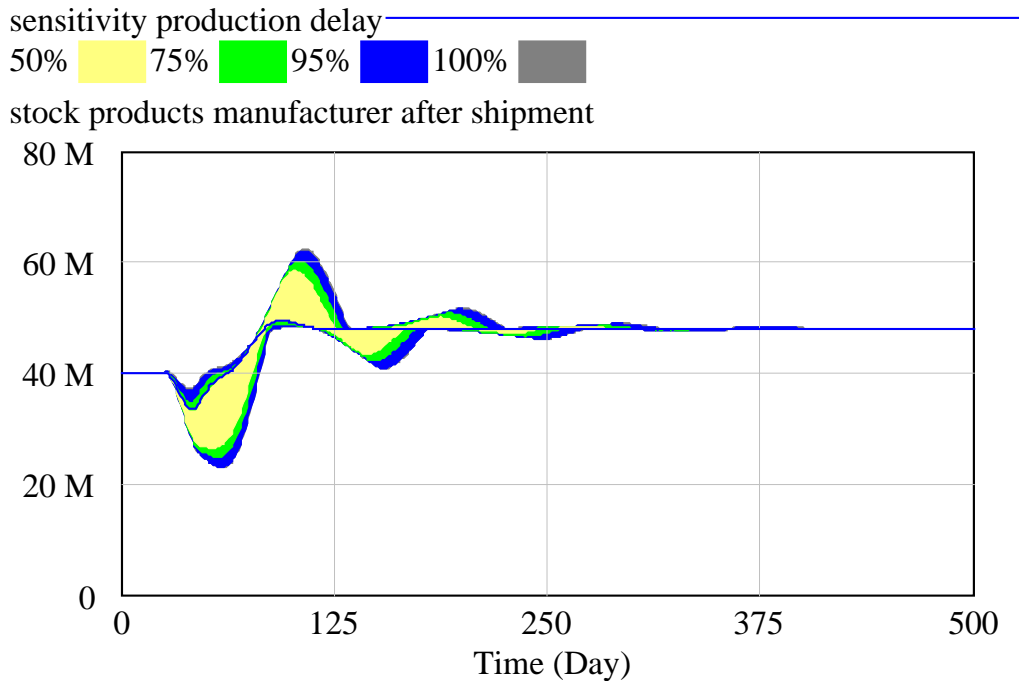


Figure 14 Sensitivity analysis for production delay (1 to 30) on the manufacturer's order rate

The system improvement test is concerned with the implementation of policies that improved the model. The real test lies in the successful implementation of these policies and the resulting real improvements in the real world. Unfortunately, the use of models in this research has been dictated by the impracticability of implementing the policies studied in the time frame available, so the system improvement test could not be carried out.

Therefore, after having undergone the twelve tests for model validation, the model studied in this thesis can be used with confidence in terms of its validity, credibility and generality.

Conclusion

It has been seen that the house building industry in the UK needs to change if it is to improve its performance. The need for a generic supply chain model for the house building industry has been recognised. The aim of this model was defined as to assess the impact of re-engineering policies on dynamic performance. It has also been seen that the development of a dynamic model is not a linear process and needs to start from simple to incorporate more complexity step by step. The model presented in this paper is based on two IOBPCS models to represent Home Builder and merchants. Other features such as unfulfilled orders, information enrichment, and mistrust were then developed and implemented. The description of the different sub-systems defined the relationships of the different variables used in the model. Several tests to build confidence in the model were reviewed and showed that the model can be used with confidence as it represents the main characteristics of the real system and is appropriate

for the purpose of the model. Finally, this model has been used to assess performance for several different scenarios of house building supply chains and the results are extensively presented and analysed in Hong-Minh (Hong-Minh, 2001).

Acknowledgments

This paper highlights work undertaken during an EPSRC IMI and DETR funded research project as part of a LINK programme on Meeting Customer Needs through Standardisation. The sponsorship and the time and effort made available by the industrial partners is gratefully acknowledged. Thanks also to Prof. D. Gann, Prof. J. Barlow, Dr. R. Ozaki, P. Childerhouse and Dr. M. Naim who were fellow research members on the project.

References

- Ball, M. (1996). Housing and Construction. A troubled relationship?, Joseph Rowntree Foundation, The Policy Press.
- Ball, M. (1999). "Chasing a snail: Innovation and housebuilding firms' strategies." Housing Studies **14**(1): 9-22.
- Barlas, Y. (1989). "Multiple test for validation of system dynamics type of simulation models." European Journal of Operation Research **42**(1): 59-87.
- Barlas, Y. (1990). "An autocorrelation function-test for output validation." Simulation **55**(1): 7-16.
- Barlas, Y. (1996). "Formal aspects of model validity and validation in system dynamics." System Dynamics Review **12**(3): 183-210.
- Barlow, J. (1998). "Take your partner for housebuilding." Agenda: the Housing Magazine(February): 16.
- Barlow, J. (1999). "From craft production to mass customisation. Innovation requirements for the UK housebuilding industry." Housing Studies **14**(1): 23-42.
- Barlow, J. (2000). "Private Sector Housebuilding. Structure and Strategies in the 21st Century."
- Berry, D. (1994). The analysis, modelling and simulation of a re-engineered PC supply chain. University of Wales Cardiff. PhD Thesis.
- Bresnen, M. (1996). Cultural Change in the Construction Industry: Developing the Clients Management Role to improve Project Performance. Proceedings of Partnering in Construction One-day conference, University of Salford and University of Westminster, 13th May, 8-20.
- Cheema (1994). Dynamic analysis of an inventory and production control system with an adaptive leadtime estimator. University of Wales Cardiff. PhD Thesis.
- Christopher, M. G. (1992). Logistics and Supply Chain Management. London, UK, Pitman Publishing.
- Churchman, C. (1973). "Reliability of models in the social sciences." Interfaces **4**(1): 1-12.
- Construction Productivity Network (1997). Practical experience of construction productivity improvement, Workshop Report, 721S: 1-13.

- Construction Productivity Network (1998). Partnering in the team: changing the culture and practice of construction, Workshop Report, 819B: 1-8.
- Coyle, R. G. (1977). Management system dynamics, John Wiley and Sons, New York.
- Disney, S. M., M. M. Naim and D. R. Towill (2000). "Generic algorithm optimisation of a class of inventory control systems." International Journal of Production Economics **68**(3): 259-278.
- Edghill, J. (1990). The application of aggregate industrial dynamics techniques to manufacturing systems. University of Wales Cardiff. PhD Thesis.
- Egan, J. S. (1998). Rethinking construction: the report of the Construction Task Force, Department of the Environment, Transport and the Regions, London.
- Ferris, J. S. and D. R. Towill (1993). "Benchmarking of a generic family of dynamic manufacturing ordering and control models." Journal of Systems Engineering **3**: 170-182.
- Forrester, J. W. (1958). "Industrial Dynamics: A major breakthrough for decision makers." Harvard Business Review **36**(4): 37-66.
- Forrester, J. W. (1961). Industrial Dynamics. Boston, MA, USA, MIT Press.
- Forrester, J. W. and P. M. Senge (1980). Test for building confidence in system dynamics models. System Dynamics. A. A. Legasto and J. W. Forrester. Amsterdam, North-Holland: 209-228.
- Gann, D. (1996). "Construction as a manufacturing process? Similarities and differences between industrialized housing and car production in Japan." Construction Management and Economics **14**: 437-450.
- Gann, D. (1998). Technology and industrial performance in construction. in OECD 'Revitalising mature industries: the role of advanced technology', Chapter 5. Paris, Science, Technology and Industry Outlook 1998, pp. 150-161.
- Gann, D. (2000). Building Innovation complex constructs in a changing world. London, Thomas Telford.
- Gann, D. and P. Senker (1998). "Construction skills training for the next millennium." Construction Management Economics **Feb**: 569-580.
- Gann, D., Y. Wang and R. Hawkins (1998). "Do regulations encourage innovation? - The case of energy efficiency in housing." Building Research & Information **26**(4): 1-17.
- Hill, T. (2000). Operations Management Strategic Context and Managerial Analysis. Chippenham, MacMillan Business.
- Holti, R. (1996). Partnering as interrelated to Technology and Organisational Change. Proceedings of Partnering in Construction one-day conference, University of Salford and University of Westminster, 13th May, pp. 59-71.
- Home Builder (2000). Data collected by the author during interviews with Home Builder's personnel.
- Hong-Minh, S., R. Barker and M. Naim (2001). "Identifying Supply Chain Solutions: those suitable for the UK house building sector." European Journal of Purchasing and Supply Management **7**(1): 49-59.
- Hong-Minh, S. M. (2001). Supply chain re-engineering strategies for the UK private housebuilding industry. Cardiff Business School, University of Wales, Cardiff. PhD Thesis. Cardiff
- Howell, G. and G. Ballard (1994). Lean Production Theory: moving beyond 'Can-Do'. Lean Construction. A. L. Rotterdam, AA Balkema, pp. 17-24.

- Jenkins, G. (1971). The system approach. System Behaviour. J. Beishon and P. G. London, OUP: 78-104.
- John, S., M. Naim and D. Towill (1994). "Dynamic analysis of a WIP compensated decision support system." International Journal of Manufacturing System Design 1(4): 283-297.
- Larson, E. (1997). "Partnering on Construction Projects: A Study of a Relationship Between Partnering Activities and Project Success." IEEE Transactions on Engineering Management 44(2): 188-195.
- Latham, M. (1994). Constructing the Team. London, HMSO.
- Lewis, J. (1997). An integrated approach to re-engineering material flow within a seamless supply chain. Cardiff University. PhD Thesis. Cardiff
- Love, P., H. Li and P. Mandal (1999). "Rework: a symptom of a dysfunctional supply chain." European Journal of Purchasing & Supply Management 5: 1-11.
- Mason-Jones, R. (1998). The holistic strategy of market information enrichment through the supply chain. Department of Maritime Studies and International Transport, University of Wales Cardiff. PhD Thesis. Cardiff
- Mason-Jones, R., M. Naim and D. Towill (1997). "The impact of pipeline control on supply chain dynamics." The International Journal of Logistics Management 8(2): 47-62.
- Meikle, J. (1999). Do we build enough houses? A review of the adequacy of UK investment in housing. Paper presented to the JRF Land Inquiry.
- Naim, M. and D. Towill (1993). Establishing a framework for effective materials logistics management. Proceedings of the 1st International Symposium on Logistics, Nottingham University, July 6-7.
- Office of Science and Technology (1995). Technology Foresight-Progress Through Partnership: Construction, Office of Science and Technology, HMSO.
- Ozaki, R. (1999). Society, Culture and Housing Form in England and Japan. Doctoral Thesis, University of Sussex.
- Roy, R. and C. P. Gaze (1999). Introduction of New Technology for a Customer-Focused Speculative House-Building Industry. Proceedings of the 15th Annual Conference of the Association of Researchers in Construction Management, Vol. 1, Liverpool, ARCOM, 85-94.
- Solberg, J. (1992). The power of simple models in manufacturing. Manufacturing systems - Foundations of World-Class Practice. J. Hein and W. Compton. Washington, USA National Academy of Engineering Press: 215-223.
- Sterman, J. (2000). Business Dynamics - Systems Thinking and Modeling for Complex World. Boston, McGraw-Hill Higher Education.
- Strohhecker, J. (1998). System- und objektorientierte Simulation betriebswirtschaftlicher Entscheidungen. Berlin, Duncker & Humblot.
- Towill, D. and A. Del Vecchio (1994). "The application of the filter theory the study of supply chain dynamics." Production Planning & Control 5(1): 82-96.
- Towill, D., G. Evans and P. Cheema (1997). "Analysis and design of an adaptive minimum reasonable inventory control system." International Journal of Production Planning and Control 8(6): 545-557.
- Towill, D. R. (1982). "Dynamic analysis of an inventory and order based production control system." International Journal of Production Research 20(6): 671-687.

van Ackere, A., E. Larsen and J. Morecroft (1993). "Systems thinking and business process redesign: an application to the beer game." European Management Journal **11**(4): 412-423.

Appendix 1 – Model documentation

The variables are grouped under the six sub-systems presented in the paper, namely Home Builder, Manufacturer, Measure of performance, Merchants, Production allocation, Site construction plus the Control variables. For simplification of presentation, the units for each variable are not shown here.

.Control

Simulation Control Parameters

FINAL TIME = 1095

INITIAL TIME = 0

SAVEPER = 1

TIME STEP = 1

.Home Builder

call off products by the site =
DELAY FIXED (coef products
* order placed by the regions ,order to call off delay ,
1e+007)

call off to delivery delay = 7

coef products = 100000

construction plan = 100
+ STEP (20, 20)

expected delivery products by the site =
DELAY FIXED (call off products by the site ,call off to delivery delay ,
1e+007)

expected product merchant1 =
DELAY FIXED (call off for products merchant1 ,call off to delivery delay ,
3e+006)

expected product merchant2 =

DELAY FIXED (call off product merchant2 ,call off to delivery delay ,
500000)

expected product merchant3 =
DELAY FIXED (call off product merchant3 ,call off to delivery delay ,
1e+006)

expected product merchant4 =
DELAY FIXED (call off product merchant4 ,call off to delivery delay ,
1e+006)

expected product merchant5 =
DELAY FIXED (call off product merchant5 ,call off to delivery delay ,
300000)

expected product merchant6 =
DELAY FIXED (call off product merchant6 ,call off to delivery delay ,
2e+006)

expected product merchant7 =
DELAY FIXED (call off product merchant7 ,call off to delivery delay ,
1.2e+006)

expected product merchant8 =
DELAY FIXED (call off product merchant8 ,call off to delivery delay ,
1e+006)

order placed by the regions =
construction plan

order to call off delay = 14

.Manufacturer

average demand products manufacturer =
SMOOTH (total demand for products manufacturer inc mistrust ,
ta products manufacturer)

completion rate products manufacturer =
DELAY1 (orate products manufacturer ,
Tp products manufacturer)

incoming backlog products manufacturer =
DELAY FIXED (total products call off manufacturer , "order to delivery to
merchants delay-1" ,
1e+007)

information enrichment coefficient = 0

inventory error manufacturer products =
target inventory products manufacturer
- stock products manufacturer after shipment
+ order backlog products manufacturer

inventory error manufacturer products delay1 =
DELAY FIXED (inventory error manufacturer products ,1,
0)

orate products manufacturer =
orders enriched manufacturer
+ inventory error manufacturer products delay1
/ Ti products manufacturer

order backlog products manufacturer =
INTEG(incoming backlog products manufacturer
- processed backlog products manufacturer ,
0)

"order to delivery to merchants delay-1" = 6

orders enriched manufacturer =
information enrichment coefficient
* average order products to merchants
+ ((1
- information enrichment coefficient)
* average demand products manufacturer)

processed backlog products manufacturer =
product to be delivered to merchants

product to be delivered to merchants =
IF THEN ELSE (stock products manufacturer before shipment
<= 0,
0,
IF THEN ELSE (stock products manufacturer before shipment
< (incoming backlog products manufacturer
+ order backlog products manufacturer) ,
stock products manufacturer before shipment ,
(incoming backlog products manufacturer
+ order backlog products manufacturer)))

stock products manufacturer after shipment =
stock products manufacturer before shipment
- product to be delivered to merchants

stock products manufacturer after shipment delay1 =
DELAY FIXED (stock products manufacturer after shipment ,1,
4e+007)

stock products manufacturer before shipment =
stock products manufacturer after shipment delay1
+ completion rate products manufacturer

ta products manufacturer = 14

target inventory products manufacturer =
4
* orders enriched manufacturer

Ti products manufacturer = 7

total demand for products manufacturer inc mistrust =
(orate products merchant 1
* (1
+ (mistrust coef 1
* mistrust value reverse products manufacturer merchant1)))
+ (orate products merchant 2
* (1
+ (mistrust coef 1
* mistrust value reverse products manufacturer merchant2)))
+ (orate products merchant 3
* (1
+ (mistrust coef 1
* mistrust value reverse products manufacturer merchant3)))
+ (orate products merchant 5
* (1
+ (mistrust coef 1
* mistrust value reverse products manufacturer merchant5)))
+ (orate products merchant 6
* (1
+ (mistrust coef 1
* mistrust value reverse products manufacturer merchant6)))
+ (orate products merchant 7
* (1
+ (mistrust coef 1
* mistrust value reverse products manufacturer merchant7)))
+ (orate products merchant 8
* (1
+ (mistrust coef 1
* mistrust value reverse product manufacturer merchant8)))
+ (orate products merchant 4
* (1

$$+ (\text{mistrust coef } 1 \\ * \text{ mistrust value reverse products manufacturer merchant4 }))$$

$$\text{total products call off manufacturer} = \\ \text{orate products merchant 1} \\ + \text{ orate products merchant 2} \\ + \text{ orate products merchant 3} \\ + \text{ orate products merchant 5} \\ + \text{ orate products merchant 6} \\ + \text{ orate products merchant 7} \\ + \text{ orate products merchant 8} \\ + \text{ orate products merchant 4}$$

$$\text{Tp products manufacturer} = 7$$

.Master model5

.Measure of performance

$$\text{Absolute error inventory} = \\ \text{ABS (inventory error manufacturer products delay1)}$$

$$\text{Absolute error order rate} = \\ \text{ABS (Manufacturer order rate for houses} \\ - \text{ order placed by the regions)}$$

$$\text{cost in stock manufacturer} = 5e-006$$

$$\text{cost in stock product merchant} = 5e-006$$

$$\text{cost out stock manufacturer} = 1e-005$$

$$\text{cost out stock product merchant} = 1e-005$$

$$\text{cost product merchant1} = \\ \text{cost in stock product merchant} \\ * \text{ stock products after shipment merchant1} \\ + \text{ cost out stock product merchant} \\ * \text{ order backlog product merchant1}$$

$$\text{cost product merchant2} = \\ \text{cost in stock product merchant} \\ * \text{ stock products after shipment merchant2} \\ + \text{ cost out stock product merchant}$$

* order backlog product merchant2

cost product merchant3 =
cost in stock product merchant
* stock products after shipment merchant3
+ cost out stock product merchant
* order backlog product merchant3

cost product merchant4 =
cost in stock product merchant
* stock products after shipment merchant4
+ cost out stock product merchant
* order backlog product merchant4

cost product merchant5 =
cost in stock product merchant
* stock products after shipment merchant5
+ cost out stock product merchant
* order backlog product merchant5

cost product merchant6 =
cost in stock product merchant
* stock products after shipment merchant6
+ cost out stock product merchant
* order backlog product merchant6

cost product merchant7 =
cost in stock product merchant
* stock products after shipment merchant7
+ cost out stock product merchant
* order backlog product merchant7

cost product merchant8 =
cost in stock product merchant
* stock products after shipment merchant8
+ cost out stock product merchant
* order backlog product merchant8

cost stock manufacturer =
cost in stock manufacturer
* stock products manufacturer after shipment
+ cost out stock manufacturer
* order backlog products manufacturer

cum diff in quantity product for merchant1 =
INTEG(diff deliveries product for merchant1 ,
0)

cum diff in quantity product for merchant2 =
INTEG(diff deliveries product for merchant2 ,
0)

cum diff in quantity product for merchant3 =
INTEG(diff deliveries product for merchant3 ,
0)

cum diff in quantity product for merchant4 =
INTEG(diff deliveries product for merchant4 ,
0)

cum diff in quantity product for merchant5 =
INTEG(diff deliveries product for merchant5 ,
0)

cum diff in quantity product for merchant6 =
INTEG(diff deliveries product for merchant6 ,
0)

cum diff in quantity product for merchant7 =
INTEG(diff deliveries product for merchant7 ,
0)

cum diff in quantity product for merchant8 =
INTEG(diff deliveries product for merchant8 ,
0)

cum diff in quantity product from merchant1 =
INTEG(diff deliveries product from merchant1 ,
0)

cum diff in quantity product from merchant2 =
INTEG(diff deliveries product from merchant2 ,
0)

cum diff in quantity product from merchant3 =
INTEG(diff deliveries product from merchant3 ,
0)

cum diff in quantity product from merchant4 =
INTEG(diff deliveries product from merchant4 ,
0)

cum diff in quantity product from merchant5 =
INTEG(diff deliveries product from merchant5 ,
0)

cum diff in quantity product from merchant6 =
INTEG(diff deliveries product from merchant6 ,
0)

cum diff in quantity product from merchant7 =
INTEG(diff deliveries product from merchant7 ,
0)

cum diff in quantity product from merchant8 =
INTEG(diff deliveries product from merchant8 ,
0)

days out of stock cum product Manufacturer =
INTEG(days out of stock product Manufacturer ,
0)

days out of stock cum product merchant1 =
INTEG(days out of stock product merchant1 ,
0)

days out of stock cum product merchant2 =
INTEG(days out of stock product merchant2 ,
0)

days out of stock cum product merchant3 =
INTEG(days out of stock product merchant3 ,
0)

days out of stock cum product merchant4 =
INTEG(days out of stock product merchant4 ,
0)

days out of stock cum product merchant5 =
INTEG(days out of stock product merchant5 ,
0)

days out of stock cum product merchant6 =
INTEG(days out of stock product merchant6 ,
0)

days out of stock cum product merchant7 =
INTEG(days out of stock product merchant7 ,
0)

days out of stock cum product merchant8 =
INTEG(days out of stock product merchant8 ,
0)

days out of stock product all merchants =
days out of stock cum product merchant1
+ days out of stock cum product merchant2
+ days out of stock cum product merchant3
+ days out of stock cum product merchant4
+ days out of stock cum product merchant5
+ days out of stock cum product merchant6
+ days out of stock cum product merchant7
+ days out of stock cum product merchant8

days out of stock product Manufacturer =
IF THEN ELSE (order backlog products manufacturer
> 0,
1,
0)

days out of stock product merchant1 =
IF THEN ELSE (order backlog product merchant1
> 0,
1,
0)

days out of stock product merchant2 =
IF THEN ELSE (order backlog product merchant2
> 0,
1,
0)

days out of stock product merchant3 =
IF THEN ELSE (order backlog product merchant3
> 0,
1,
0)

days out of stock product merchant4 =
IF THEN ELSE (order backlog product merchant4
> 0,
1,
0)

days out of stock product merchant5 =
IF THEN ELSE (order backlog product merchant5
> 0.01,
1,
0)

days out of stock product merchant6 =
IF THEN ELSE (order backlog product merchant6

> 0,
1,
0)

days out of stock product merchant7 =
IF THEN ELSE (order backlog product merchant7
> 0,
1,
0)

days out of stock product merchant8 =
IF THEN ELSE (order backlog product merchant8
> 0,
1,
0)

diff deliveries product for merchant1 =
product deliveries merchant1
- order product for merchant1

diff deliveries product for merchant2 =
product deliveries merchant2
- order product for merchant2

diff deliveries product for merchant3 =
product deliveries merchant3
- order product for merchant3

diff deliveries product for merchant4 =
product deliveries merchant4
- order product for merchant4

diff deliveries product for merchant5 =
product deliveries merchant5
- order product for merchant5

diff deliveries product for merchant6 =
product deliveries merchant6
- order product for merchant6

diff deliveries product for merchant7 =
product deliveries merchant7
- order product for merchant7

diff deliveries product for merchant8 =
product deliveries merchant8
- order product for merchant8

diff deliveries product from merchant1 =
product to be delivered from merchant1
- incoming backlog product merchant1

diff deliveries product from merchant2 =
product to be delivered from merchant2
- incoming backlog product merchant2

diff deliveries product from merchant3 =
product to be delivered from merchant3
- incoming backlog product merchant3

diff deliveries product from merchant4 =
product to be delivered from merchant4
- incoming backlog product merchant4

diff deliveries product from merchant5 =
product to be delivered from merchant5
- incoming backlog product merchant5

diff deliveries product from merchant6 =
product to be delivered from merchant6
- incoming backlog product merchant6

diff deliveries product from merchant7 =
product to be delivered from merchant7
- incoming backlog product merchant7

diff deliveries product from merchant8 =
product to be delivered from merchant8
- incoming backlog product merchant8

Integrated absolute error inventory =
INTEG(Absolute error inventory ,
0)

Integrated absolute error order rate =
INTEG(Absolute error order rate ,
0)

Manufacturer order rate for houses =
orate products manufacturer
/ coef products

mistrust value product manufacturer merchant1 =
ZIDZ (diff deliveries product for merchant1 ,
order product for merchant1)

mistrust value product manufacturer merchant2 =
ZIDZ (diff deliveries product for merchant2 ,
order product for merchant2)

mistrust value product manufacturer merchant3 =
ZIDZ (diff deliveries product for merchant3 ,
order product for merchant3)

mistrust value product manufacturer merchant4 =
ZIDZ (diff deliveries product for merchant4 ,
order product for merchant4)

mistrust value product manufacturer merchant5 =
ZIDZ (diff deliveries product for merchant5 ,
order product for merchant5)

mistrust value product manufacturer merchant6 =
ZIDZ (diff deliveries product for merchant6 ,
order product for merchant6)

mistrust value product manufacturer merchant7 =
ZIDZ (diff deliveries product for merchant7 ,
order product for merchant7)

mistrust value product manufacturer merchant8 =
ZIDZ (diff deliveries product for merchant8 ,
order product for merchant8)

mistrust value product merchant1 =
ZIDZ (diff deliveries product from merchant1 ,
order product from merchant1)

mistrust value product merchant2 =
ZIDZ (diff deliveries product from merchant2 ,
order product from merchant2)

mistrust value product merchant3 =
ZIDZ (diff deliveries product from merchant3 ,
order product from merchant3)

mistrust value product merchant4 =
ZIDZ (diff deliveries product from merchant4 ,
order product from merchant4)

mistrust value product merchant5 =
ZIDZ (diff deliveries product from merchant5 ,
order product from merchant5)

mistrust value product merchant6 =
ZIDZ (diff deliveries product from merchant6 ,
order product from merchant6)

mistrust value product merchant7 =
ZIDZ (diff deliveries product from merchant7 ,
order product from merchant7)

mistrust value product merchant8 =
ZIDZ (diff deliveries product from merchant8 ,
order product from merchant8)

mistrust value reverse product manufacturer merchant8 =
DELAY FIXED ((mistrust value product manufacturer merchant8
* (-1)) ,1,
0)

mistrust value reverse product merchant1 =
DELAY FIXED ((mistrust value product merchant1
* (-1)) ,1,
0)

mistrust value reverse product merchant2 =
DELAY FIXED ((mistrust value product merchant2
* (-1)) ,1,
0)

mistrust value reverse product merchant3 =
DELAY FIXED ((mistrust value product merchant3
* (-1)) ,1,
0)

mistrust value reverse product merchant4 =
DELAY FIXED ((mistrust value product merchant4
* (-1)) ,1,
0)

mistrust value reverse product merchant5 =
DELAY FIXED ((mistrust value product merchant5
* (-1)) ,1,
0)

mistrust value reverse product merchant6 =
DELAY FIXED ((mistrust value product merchant6
* (-1)) ,1,
0)

mistrust value reverse product merchant7 =

DELAY FIXED ((mistrust value product merchant7
* (-1)),1,
0)

mistrust value reverse product merchant8 =
DELAY FIXED ((mistrust value product merchant8
* (-1)),1,
0)

mistrust value reverse products manufacturer merchant1 =
DELAY FIXED ((mistrust value product manufacturer merchant1
* (-1)),1,
0)

mistrust value reverse products manufacturer merchant2 =
DELAY FIXED ((mistrust value product manufacturer merchant2
* (-1)),1,
0)

mistrust value reverse products manufacturer merchant3 =
DELAY FIXED ((mistrust value product manufacturer merchant3
* (-1)),1,
0)

mistrust value reverse products manufacturer merchant4 =
DELAY FIXED ((mistrust value product manufacturer merchant4
* (-1)),1,
0)

mistrust value reverse products manufacturer merchant5 =
DELAY FIXED ((mistrust value product manufacturer merchant5
* (-1)),1,
0)

mistrust value reverse products manufacturer merchant6 =
DELAY FIXED ((mistrust value product manufacturer merchant6
* (-1)),1,
0)

mistrust value reverse products manufacturer merchant7 =
DELAY FIXED ((mistrust value product manufacturer merchant7
* (-1)),1,
0)

order product for merchant1 =
DELAY FIXED (orate products merchant 1 ,9,
3e+006)

order product for merchant2 =
DELAY FIXED (orate products merchant 2 ,9,
500000)

order product for merchant3 =
DELAY FIXED (orate products merchant 3 ,9,
1e+006)

order product for merchant4 =
DELAY FIXED (orate products merchant 4 ,9,
1e+006)

order product for merchant5 =
DELAY FIXED (orate products merchant 5 ,9,
300000)

order product for merchant6 =
DELAY FIXED (orate products merchant 6 ,9,
2e+006)

order product for merchant7 =
DELAY FIXED (orate products merchant 7 ,9,
1.2e+006)

order product for merchant8 =
DELAY FIXED (orate products merchant 8 ,9,
1e+006)

order product from merchant1 =
incoming backlog product merchant1

order product from merchant2 =
incoming backlog product merchant2

order product from merchant3 =
incoming backlog product merchant3

order product from merchant4 =
incoming backlog product merchant4

order product from merchant5 =
incoming backlog product merchant5

order product from merchant6 =
incoming backlog product merchant6

order product from merchant7 =
incoming backlog product merchant7

order product from merchant8 =
incoming backlog product merchant8

"Production on-costs" =
Integrated absolute error order rate
* Integrated absolute error order rate
* Integrated absolute error order rate

SC costs =
stock cost manufacturer cum
+ stock costs product for all merchants

stock cost manufacturer cum =
INTEG(cost stock manufacturer ,
0)

stock cost product merchant1 =
INTEG(cost product merchant1 ,
0)

stock cost product merchant2 =
INTEG(cost product merchant2 ,
0)

stock cost product merchant3 =
INTEG(cost product merchant3 ,
0)

stock cost product merchant4 =
INTEG(cost product merchant4 ,
0)

stock cost product merchant5 =
INTEG(cost product merchant5 ,
0)

stock cost product merchant6 =
INTEG(cost product merchant6 ,
0)

stock cost product merchant7 =
INTEG(cost product merchant7 ,
0)

stock cost product merchant8 =
INTEG(cost product merchant8 ,
0)

stock costs product for all merchants =
stock cost product merchant1
+ stock cost product merchant2
+ stock cost product merchant3
+ stock cost product merchant4
+ stock cost product merchant5
+ stock cost product merchant6
+ stock cost product merchant7
+ stock cost product merchant8

sum expected deliveries product for merchant1 =
INTEG(order product for merchant1 ,
0)

sum expected deliveries product for merchant2 =
INTEG(order product for merchant2 ,
0)

sum expected deliveries product for merchant3 =
INTEG(order product for merchant3 ,
0)

sum expected deliveries product for merchant4 =
INTEG(order product for merchant4 ,
0)

sum expected deliveries product for merchant5 =
INTEG(order product for merchant5 ,
0)

sum expected deliveries product for merchant6 =
INTEG(order product for merchant6 ,
0)

sum expected deliveries product for merchant7 =
INTEG(order product for merchant7 ,
0)

sum expected deliveries product for merchant8 =
INTEG(order product for merchant8 ,
0)

sum expected deliveries product from merchant1 =
INTEG(order product from merchant1 ,
0)

sum expected deliveries product from merchant2 =

INTEG(order product from merchant2 ,
0)

sum expected deliveries product from merchant3 =
INTEG(order product from merchant3 ,
0)

sum expected deliveries product from merchant4 =
INTEG(order product from merchant4 ,
0)

sum expected deliveries product from merchant5 =
INTEG(order product from merchant5 ,
0)

sum expected deliveries product from merchant6 =
INTEG(order product from merchant6 ,
0)

sum expected deliveries product from merchant7 =
INTEG(order product from merchant7 ,
0)

sum expected deliveries product from merchant8 =
INTEG(order product from merchant8 ,
0)

.Merchants

average order products to merchant 1 =
SMOOTH (order product inc mistrust merchant1 ,
Ta products merchant)

average order products to merchant 2 =
SMOOTH (order product inc mistrust merchant2 ,
Ta products merchant)

average order products to merchant 3 =
SMOOTH (order product inc mistrust merchant3 ,
Ta products merchant)

average order products to merchant 4 =
SMOOTH (order product inc mistrust merchant4 ,
Ta products merchant)

average order products to merchant 5 =

SMOOTH (order product inc mistrust merchant5 ,
Ta products merchant)

average order products to merchant 6 =
SMOOTH (order product inc mistrust merchant6 ,
Ta products merchant)

average order products to merchant 7 =
SMOOTH (order product inc mistrust merchant7 ,
Ta products merchant)

average order products to merchant 8 =
SMOOTH (order product inc mistrust merchant8 ,
Ta products merchant)

average order products to merchants =
average order products to merchant 1
+ average order products to merchant 2
+ average order products to merchant 3
+ average order products to merchant 4
+ average order products to merchant 5
+ average order products to merchant 6
+ average order products to merchant 7
+ average order products to merchant 8

call off for products merchant1 =
call off products by the site
* coef product merchant1

call off product merchant2 =
call off products by the site
* coef product merchant2

call off product merchant3 =
call off products by the site
* coef product merchant3

call off product merchant4 =
call off products by the site
* coef product merchant4

call off product merchant5 =
call off products by the site
* coef product merchant5

call off product merchant6 =
call off products by the site
* coef product merchant6

call off product merchant7 =
call off products by the site
* coef product merchant7

call off product merchant8 =
call off products by the site
* coef product merchant8

"call off to delivery delay-1" = 6

coef product merchant1 = 0.3

coef product merchant2 = 0.05

coef product merchant3 = 0.1

coef product merchant4 = 0.1

coef product merchant5 = 0.03

coef product merchant6 = 0.2

coef product merchant7 = 0.12

coef product merchant8 = 0.1

incoming backlog product merchant1 =
DELAY FIXED (call off for products merchant1 ,"call off to delivery delay-1" ,
3e+006)

incoming backlog product merchant2 =
DELAY FIXED (call off product merchant2 ,"call off to delivery delay-1" ,
500000)

incoming backlog product merchant3 =
DELAY FIXED (call off product merchant3 ,"call off to delivery delay-1" ,
1e+006)

incoming backlog product merchant4 =
DELAY FIXED (call off product merchant4 ,"call off to delivery delay-1" ,
1e+006)

incoming backlog product merchant5 =
DELAY FIXED (call off product merchant5 ,"call off to delivery delay-1" ,
300000)

incoming backlog product merchant6 =

DELAY FIXED (call off product merchant6 ,"call off to delivery delay-1" ,
2e+006)

incoming backlog product merchant7 =
DELAY FIXED (call off product merchant7 ,"call off to delivery delay-1" ,
1.2e+006)

incoming backlog product merchant8 =
DELAY FIXED (call off product merchant8 ,"call off to delivery delay-1" ,
1e+006)

incoming products merchant 1 =
DELAY FIXED (product deliveries merchant1 ,transport delay ,
3e+006)

incoming products merchant2 =
DELAY FIXED (product deliveries merchant2 ,transport delay ,
500000)

incoming products merchant3 =
DELAY FIXED (product deliveries merchant3 ,transport delay ,
1e+006)

incoming products merchant4 =
DELAY FIXED (product deliveries merchant4 ,transport delay ,
1e+006)

incoming products merchant5 =
DELAY FIXED (product deliveries merchant5 ,transport delay ,
300000)

incoming products merchant6 =
DELAY FIXED (product deliveries merchant6 ,transport delay ,
2e+006)

incoming products merchant7 =
DELAY FIXED (product deliveries merchant7 ,transport delay ,
1.2e+006)

incoming products merchant8 =
DELAY FIXED (product deliveries merchant8 ,transport delay ,
1e+006)

inventory error product merchant1 =
"target inv. product merchant1"
- stock products after shipment merchant1
+ order backlog product merchant1

inventory error product merchant1 delay 1 =
DELAY FIXED (inventory error product merchant1 ,1,
0)

inventory error product merchant2 =
"target inv. product merchant2"
- stock products after shipment merchant2
+ order backlog product merchant2

inventory error product merchant2 delay 1 =
DELAY FIXED (inventory error product merchant2 ,1,
0)

inventory error product merchant3 =
"target inv. product merchant3"
- stock products after shipment merchant3
+ order backlog product merchant3

inventory error product merchant3 delay 1 =
DELAY FIXED (inventory error product merchant3 ,1,
0)

inventory error product merchant4 =
"target inv. product merchant4"
- stock products after shipment merchant4
+ order backlog product merchant4

inventory error product merchant4 delay 1 =
DELAY FIXED (inventory error product merchant4 ,1,
0)

inventory error product merchant5 =
"target inv. product merchant5"
- stock products after shipment merchant5
+ order backlog product merchant5

inventory error product merchant5 delay 1 =
DELAY FIXED (inventory error product merchant5 ,1,
0)

inventory error product merchant6 =
"target inv. product merchant6"
- stock products after shipment merchant6
+ order backlog product merchant6

inventory error product merchant6 delay 1 =
DELAY FIXED (inventory error product merchant6 ,1,
0)

inventory error product merchant7 =
"target inv. product merchant7"
- stock products after shipment merchant7
+ order backlog product merchant7

inventory error product merchant7 delay 1 =
DELAY FIXED (inventory error product merchant7 ,1,
0)

inventory error product merchant8 =
"target inv. product merchant8"
- stock products after shipment merchant8
+ order backlog product merchant8

inventory error product merchant8 delay 1 =
DELAY FIXED (inventory error product merchant8 ,1,
0)

mistrust coef 1 = 1

orate products merchant 1 =
average order products to merchant 1
+ inventory error product merchant1 delay 1
/ Ti products merchant

orate products merchant 2 =
average order products to merchant 2
+ inventory error product merchant2 delay 1
/ Ti products merchant

orate products merchant 3 =
average order products to merchant 3
+ inventory error product merchant3 delay 1
/ Ti products merchant

orate products merchant 4 =
average order products to merchant 4
+ inventory error product merchant4 delay 1
/ Ti products merchant

orate products merchant 5 =
average order products to merchant 5
+ inventory error product merchant5 delay 1
/ Ti products merchant

orate products merchant 6 =
average order products to merchant 6

+ inventory error product merchant6 delay 1
/ Ti products merchant

orate products merchant 7 =
average order products to merchant 7
+ inventory error product merchant7 delay 1
/ Ti products merchant

orate products merchant 8 =
average order products to merchant 8
+ inventory error product merchant8 delay 1
/ Ti products merchant

order backlog product merchant1 =
INTEG(incoming backlog product merchant1
- processed backlog product merchant 1 ,
0)

order backlog product merchant2 =
INTEG(incoming backlog product merchant2
- processed backlog product merchant 2 ,
0)

order backlog product merchant3 =
INTEG(incoming backlog product merchant3
- processed backlog product merchant 3 ,
0)

order backlog product merchant4 =
INTEG(incoming backlog product merchant4
- processed backlog product merchant 4 ,
0)

order backlog product merchant5 =
INTEG(incoming backlog product merchant5
- processed backlog product merchant 5 ,
0)

order backlog product merchant6 =
INTEG(incoming backlog product merchant6
- processed backlog product merchant 6 ,
0)

order backlog product merchant7 =
INTEG(incoming backlog product merchant7
- processed backlog product merchant 7 ,
0)

order backlog product merchant8 =
INTEG(incoming backlog product merchant8
- processed backlog product merchant 8 ,
0)

order for products =
order placed by the regions
* coef products

order product inc mistrust merchant1 =
order product merchant 1
* (1
+ (mistrust coef 1
* mistrust value reverse product merchant1))

order product inc mistrust merchant2 =
order product merchant2
* (1
+ (mistrust coef 1
* mistrust value reverse product merchant2))

order product inc mistrust merchant3 =
order product merchant3
* (1
+ (mistrust coef 1
* mistrust value reverse product merchant3))

order product inc mistrust merchant4 =
order product merchant4
* (1
+ (mistrust coef 1
* mistrust value reverse product merchant4))

order product inc mistrust merchant5 =
order product merchant5
* (1
+ (mistrust coef 1
* mistrust value reverse product merchant5))

order product inc mistrust merchant6 =
order product merchant6
* (1
+ (mistrust coef 1
* mistrust value reverse product merchant6))

order product inc mistrust merchant7 =
order product merchant7
* (1

$$+ (\text{mistrust coef 1} \\ * \text{mistrust value reverse product merchant7}))$$

$$\text{order product inc mistrust merchant8} = \\ \text{order product merchant8} \\ * (1 \\ + (\text{mistrust coef 1} \\ * \text{mistrust value reverse product merchant8}))$$

$$\text{order product merchant 1} = \\ \text{order for products} \\ * \text{coef product merchant1}$$

$$\text{order product merchant2} = \\ \text{order for products} \\ * \text{coef product merchant2}$$

$$\text{order product merchant3} = \\ \text{order for products} \\ * \text{coef product merchant3}$$

$$\text{order product merchant4} = \\ \text{order for products} \\ * \text{coef product merchant4}$$

$$\text{order product merchant5} = \\ \text{order for products} \\ * \text{coef product merchant5}$$

$$\text{order product merchant6} = \\ \text{order for products} \\ * \text{coef product merchant6}$$

$$\text{order product merchant7} = \\ \text{order for products} \\ * \text{coef product merchant7}$$

$$\text{order product merchant8} = \\ \text{order for products} \\ * \text{coef product merchant8}$$

$$\text{processed backlog product merchant 1} = \\ \text{product to be delivered from merchant1}$$

$$\text{processed backlog product merchant 2} = \\ \text{product to be delivered from merchant2}$$

$$\text{processed backlog product merchant 3} =$$

product to be delivered from merchant3

processed backlog product merchant 4 =
product to be delivered from merchant4

processed backlog product merchant 5 =
product to be delivered from merchant5

processed backlog product merchant 6 =
product to be delivered from merchant6

processed backlog product merchant 7 =
product to be delivered from merchant7

processed backlog product merchant 8 =
product to be delivered from merchant8

product deliveries merchant1 delay1 =
DELAY FIXED (product to be delivered from merchant1 ,1,
3e+006)

product deliveries merchant2 delay1 =
DELAY FIXED (product to be delivered from merchant2 ,1,
500000)

product deliveries merchant3 delay1 =
DELAY FIXED (product to be delivered from merchant3 ,1,
1e+006)

product deliveries merchant4 delay1 =
DELAY FIXED (product to be delivered from merchant4 ,1,
1e+006)

product deliveries merchant5 delay1 =
DELAY FIXED (product to be delivered from merchant5 ,1,
300000)

product deliveries merchant6 delay1 =
DELAY FIXED (product to be delivered from merchant6 ,1,
2e+006)

product deliveries merchant7 delay1 =
DELAY FIXED (product to be delivered from merchant7 ,1,
1.2e+006)

product deliveries merchant8 delay1 =
DELAY FIXED (product to be delivered from merchant8 ,1,
1e+006)

product deliveries on site =
product deliveries merchant1 delay1
+ product deliveries merchant2 delay1
+ product deliveries merchant3 delay1
+ product deliveries merchant4 delay1
+ product deliveries merchant5 delay1
+ product deliveries merchant6 delay1
+ product deliveries merchant7 delay1
+ product deliveries merchant8 delay1

product to be delivered from merchant1 =
IF THEN ELSE (stock products before shipment merchant 1
 <= 0,
0,
IF THEN ELSE (stock products before shipment merchant 1
 < (incoming backlog product merchant1
 + order backlog product merchant1) ,
stock products before shipment merchant 1 ,
 (incoming backlog product merchant1
 + order backlog product merchant1)))

product to be delivered from merchant2 =
IF THEN ELSE (stock products before shipment merchant 2
 <= 0,
0,
IF THEN ELSE (stock products before shipment merchant 2
 < (incoming backlog product merchant2
 + order backlog product merchant2) ,
stock products before shipment merchant 2 ,
 (incoming backlog product merchant2
 + order backlog product merchant2)))

product to be delivered from merchant3 =
IF THEN ELSE (stock products before shipment merchant 3
 <= 0,
0,
IF THEN ELSE (stock products before shipment merchant 3
 < (incoming backlog product merchant3
 + order backlog product merchant3) ,
stock products before shipment merchant 3 ,
 (incoming backlog product merchant3
 + order backlog product merchant3)))

product to be delivered from merchant4 =
IF THEN ELSE (stock products before shipment merchant 4
 <= 0,
0,

IF THEN ELSE (stock products before shipment merchant 4
 < (incoming backlog product merchant4
 + order backlog product merchant4) ,
stock products before shipment merchant 4 ,
 (incoming backlog product merchant4
 + order backlog product merchant4)))

product to be delivered from merchant5 =

IF THEN ELSE (stock products before shipment merchant 5
 <= 0,
0,
IF THEN ELSE (stock products before shipment merchant 5
 < (incoming backlog product merchant5
 + order backlog product merchant5) ,
stock products before shipment merchant 5 ,
 (incoming backlog product merchant5
 + order backlog product merchant5)))

product to be delivered from merchant6 =

IF THEN ELSE (stock products before shipment merchant 6
 <= 0,
0,
IF THEN ELSE (stock products before shipment merchant 6
 < (incoming backlog product merchant6
 + order backlog product merchant6) ,
stock products before shipment merchant 6 ,
 (incoming backlog product merchant6
 + order backlog product merchant6)))

product to be delivered from merchant7 =

IF THEN ELSE (stock products before shipment merchant 7
 <= 0,
0,
IF THEN ELSE (stock products before shipment merchant 7
 < (incoming backlog product merchant7
 + order backlog product merchant7) ,
stock products before shipment merchant 7 ,
 (incoming backlog product merchant7
 + order backlog product merchant7)))

product to be delivered from merchant8 =

IF THEN ELSE (stock products before shipment merchant8
 <= 0,
0,
IF THEN ELSE (stock products before shipment merchant8
 < (incoming backlog product merchant8
 + order backlog product merchant8) ,
stock products before shipment merchant8 ,

(incoming backlog product merchant8
+ order backlog product merchant8))

stock product after shipment merchant1 delay1 =
DELAY FIXED (stock products after shipment merchant1 ,1,
1.2e+007)

stock product after shipment merchant2 delay1 =
DELAY FIXED (stock products after shipment merchant2 ,1,
2e+006)

stock product after shipment merchant3 delay1 =
DELAY FIXED (stock products after shipment merchant3 ,1,
4e+006)

stock product after shipment merchant4 delay1 =
DELAY FIXED (stock products after shipment merchant4 ,1,
4e+006)

stock product after shipment merchant5 delay1 =
DELAY FIXED (stock products after shipment merchant5 ,1,
1.2e+006)

stock product after shipment merchant6 delay1 =
DELAY FIXED (stock products after shipment merchant6 ,1,
8e+006)

stock product after shipment merchant7 delay1 =
DELAY FIXED (stock products after shipment merchant7 ,1,
4.8e+006)

stock product after shipment merchant8 delay1 =
DELAY FIXED (stock products after shipment merchant8 ,1,
4e+006)

stock products after shipment merchant1 =
stock products before shipment merchant 1
- product to be delivered from merchant1

stock products after shipment merchant2 =
stock products before shipment merchant 2
- product to be delivered from merchant2

stock products after shipment merchant3 =
stock products before shipment merchant 3
- product to be delivered from merchant3

stock products after shipment merchant4 =

stock products before shipment merchant 4
- product to be delivered from merchant4

stock products after shipment merchant5 =
stock products before shipment merchant 5
- product to be delivered from merchant5

stock products after shipment merchant6 =
stock products before shipment merchant 6
- product to be delivered from merchant6

stock products after shipment merchant7 =
stock products before shipment merchant 7
- product to be delivered from merchant7

stock products after shipment merchant8 =
stock products before shipment merchant8
- product to be delivered from merchant8

stock products before shipment merchant 1 =
incoming products merchant 1
+ stock product after shipment merchant1 delay1

stock products before shipment merchant 2 =
incoming products merchant2
+ stock product after shipment merchant2 delay1

stock products before shipment merchant 3 =
incoming products merchant3
+ stock product after shipment merchant3 delay1

stock products before shipment merchant 4 =
incoming products merchant4
+ stock product after shipment merchant4 delay1

stock products before shipment merchant 5 =
incoming products merchant5
+ stock product after shipment merchant5 delay1

stock products before shipment merchant 6 =
incoming products merchant6
+ stock product after shipment merchant6 delay1

stock products before shipment merchant 7 =
incoming products merchant7
+ stock product after shipment merchant7 delay1

stock products before shipment merchant8 =

incoming products merchant8
+ stock product after shipment merchant8 delay1

Ta products merchant = 14

"target inv. product merchant1" =
4
* average order products to merchant 1

"target inv. product merchant2" =
4
* average order products to merchant 2

"target inv. product merchant3" =
4
* average order products to merchant 3

"target inv. product merchant4" =
4
* average order products to merchant 4

"target inv. product merchant5" =
4
* average order products to merchant 5

"target inv. product merchant6" =
4
* average order products to merchant 6

"target inv. product merchant7" =
4
* average order products to merchant 7

"target inv. product merchant8" =
4
* average order products to merchant 8

Ti products merchant = 14

transport delay = 1

wip products merchant 1 =
INTEG(orate products merchant 1
- incoming products merchant 1 ,
2.1e+007)

wip products merchant 2 =
INTEG(orate products merchant 2

- incoming products merchant2 ,
3.5e+006)

wip products merchant3 =
INTEG(orate products merchant 3
- incoming products merchant3 ,
7e+006)

wip products merchant4 =
INTEG(orate products merchant 4
- incoming products merchant4 ,
7e+006)

wip products merchant5 =
INTEG(orate products merchant 5
- incoming products merchant5 ,
2.1e+006)

wip products merchant6 =
INTEG(orate products merchant 6
- incoming products merchant6 ,
1.4e+007)

wip products merchant7 =
INTEG(orate products merchant 7
- incoming products merchant7 ,
8.4e+006)

wip products merchant8 =
INTEG(orate products merchant 8
- incoming products merchant8 ,
7e+006)

.Production allocation

coef product merchants[merchant product 1] =
coef product merchant1

coef product merchants[merchant product 2] =
coef product merchant2

coef product merchants[merchant product 3] =
coef product merchant3

coef product merchants[merchant product 4] =
coef product merchant4

coef product merchants[merchant product 5] =
coef product merchant5

coef product merchants[merchant product 6] =

coef product merchant6
coef product merchants[merchant product 7] =
coef product merchant7
coef product merchants[merchant product 8] =
coef product merchant8

manufacturer product production allocation[Merchants product] =
allocate by priority (orate product merchant[Merchants product] ,
coef product merchants[Merchants product] ,
8,
0.01,
product to be delivered to merchants)

Merchants product : merchant product 1,merchant product 2,merchant product
3,merchant product 4,merchant product 5,merchant product 6,merchant product
7,merchant product 8

orate product merchant[merchant product 1] =
DELAY FIXED (orate products merchant 1 ,6,
3e+006)

orate product merchant[merchant product 2] =
DELAY FIXED (orate products merchant 2 ,6,
500000)

orate product merchant[merchant product 3] =
DELAY FIXED (orate products merchant 3 ,6,
1e+006)

orate product merchant[merchant product 4] =
DELAY FIXED (orate products merchant 4 ,6,
1e+006)

orate product merchant[merchant product 5] =
DELAY FIXED (orate products merchant 5 ,6,
300000)

orate product merchant[merchant product 6] =
DELAY FIXED (orate products merchant 6 ,6,
2e+006)

orate product merchant[merchant product 7] =
DELAY FIXED (orate products merchant 7 ,6,
1.2e+006)

orate product merchant[merchant product 8] =
DELAY FIXED (orate products merchant 8 ,6,
1e+006)

product deliveries merchant1 =
manufacturer product production allocation[merchant product 1]

product deliveries merchant2 =
manufacturer product production allocation[merchant product 2]

product deliveries merchant3 =
 manufacturer product production allocation[merchant product 3]

product deliveries merchant4 =
 manufacturer product production allocation[merchant product 4]

product deliveries merchant5 =
 manufacturer product production allocation[merchant product 5]

product deliveries merchant6 =
 manufacturer product production allocation[merchant product 6]

product deliveries merchant7 =
 manufacturer product production allocation[merchant product 7]

product deliveries merchant8 =
 manufacturer product production allocation[merchant product 8]

 .Site construction

coef labour = 3

construct house =
 MIN (INTEGER (stock of products to be used
 / coef products) ,
 INTEGER (stock of labour to be used
 / coef labour))

construction time delay = 63

demand for labour =
 DELAY FIXED ((call off products by the site
 / coef products)
 * coef labour ,call off to delivery delay ,
 300)

free labour =
 water proofed house
 * coef labour

incoming call off for labour =
 demand for labour

labour backlog =
 INTEG(incoming call off for labour
 - processed call off for labour ,

0)

labour to be used =
IF THEN ELSE (subcontractor available before used
 <= 0,
0,
IF THEN ELSE (subcontractor available before used
 < (incoming call off for labour
 + labour backlog) ,
subcontractor available before used ,
(incoming call off for labour
 + labour backlog)))

labour used =
construct house
* coef labour

processed call off for labour =
labour to be used

products delivered on site =
product deliveries on site

products used =
construct house
* coef products

stock of labour to be used =
INTEG(labour to be used
 - labour used ,
300)

stock of products to be used =
INTEG(products delivered on site
 - products used ,
1e+007)

sub contractor available after used =
subcontractor available before used
- labour to be used

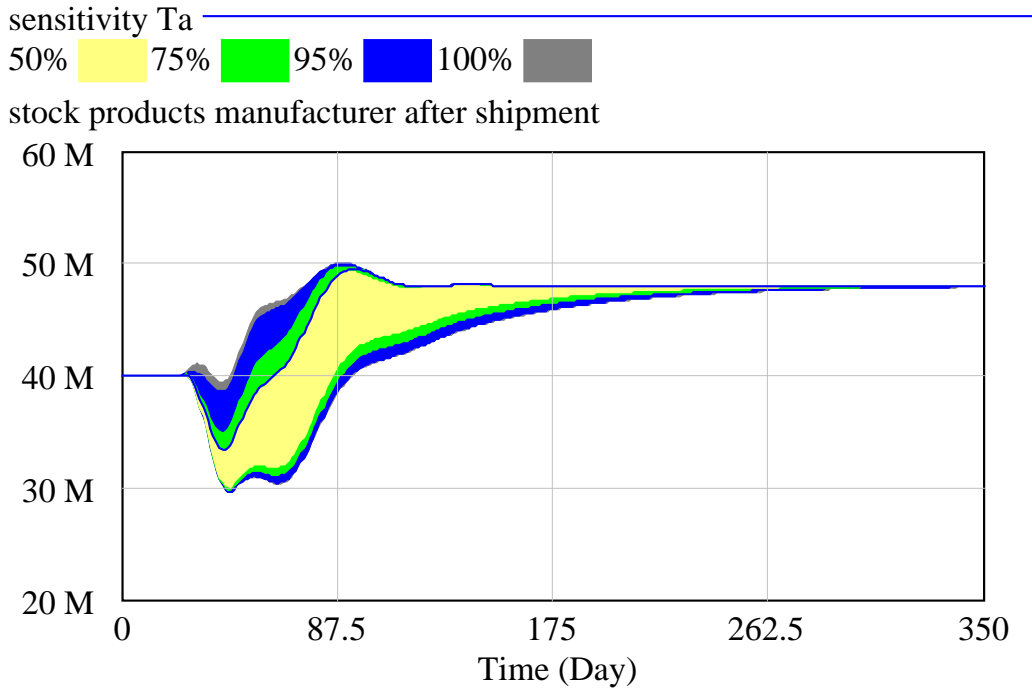
subcontractor available after delay1 =
DELAY FIXED (sub contractor available after used ,1,
5000)

subcontractor available before used =
free labour
+ subcontractor available after delay1

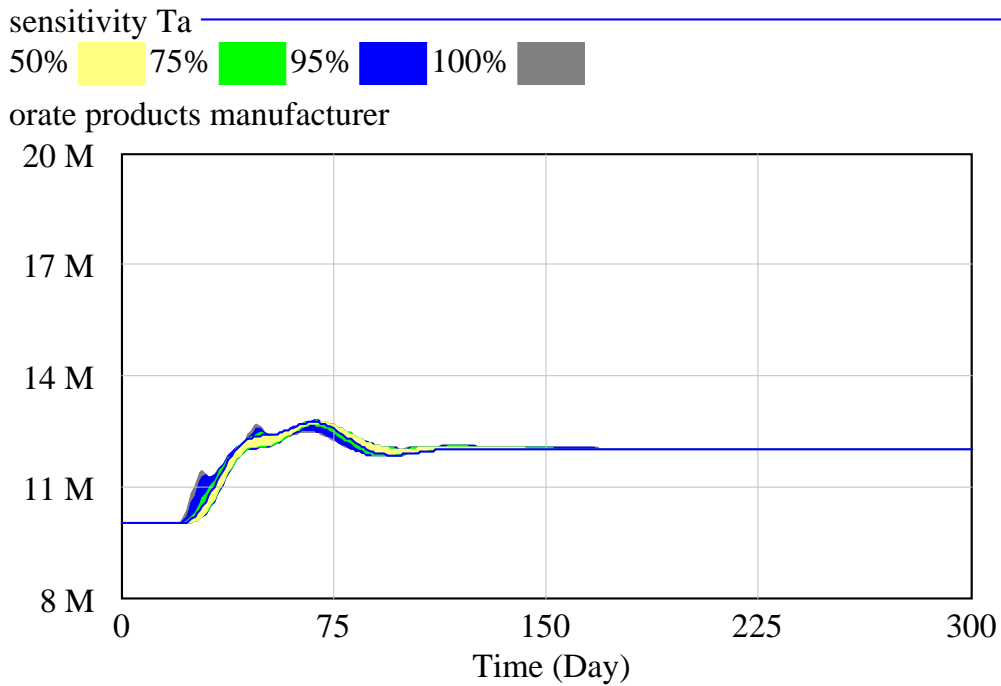
total weather proofed houses built =
INTEG(water proofed house ,
0)

water proofed house =
DELAY FIXED (construct house ,construction time delay ,
100)

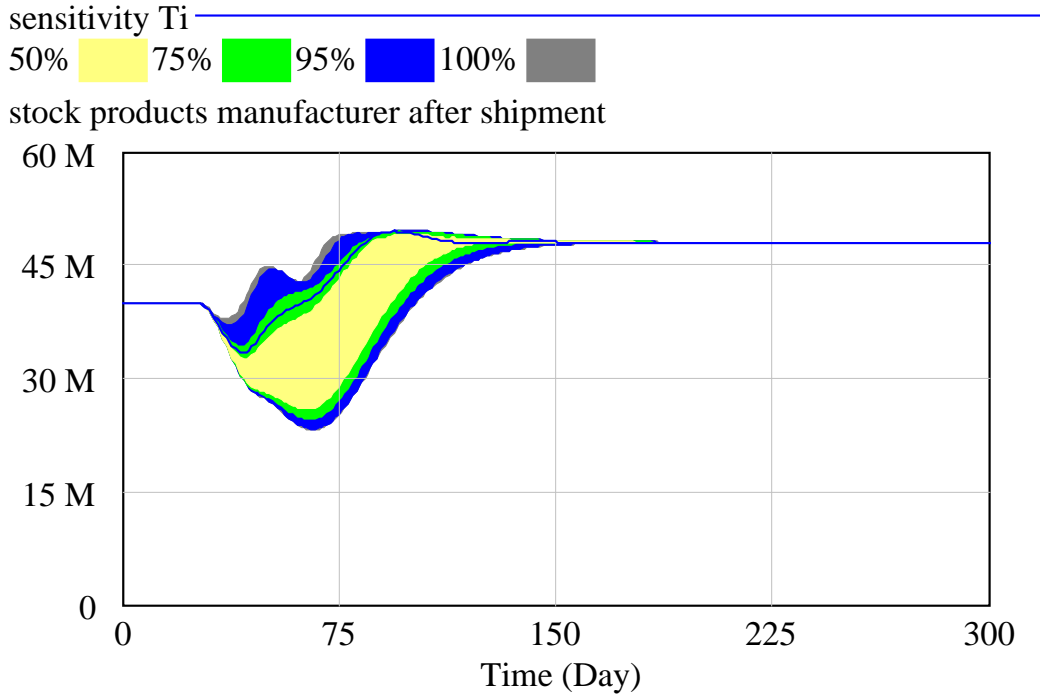
Appendix 2 – Sensitivity analysis



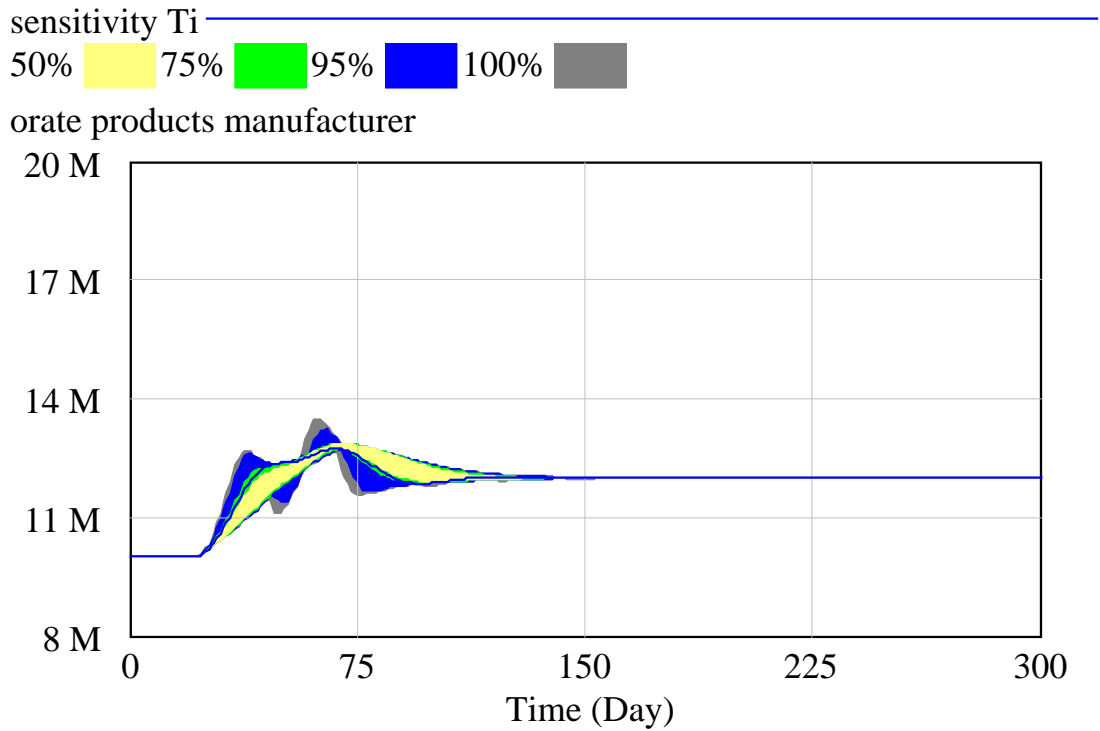
Sensitivity analysis for T_a (time to average consumption)(1 to 60) on the manufacturer's inventory level



Sensitivity analysis for T_a (time to average consumption)(1 to 60) on the manufacturer's order rate



Sensitivity analysis for T_i (time to adjust inventory)(2 to 30) on the manufacturer's inventory level



Sensitivity analysis for T_i (time to adjust inventory)(2 to 30) on the manufacturer's order rate