Effective monitoring and control with intelligent products
Meyer, Gerben Gerald

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2011

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.
Chapter 4

System Prototype for Production

As discussed in the previous chapter, centralised production planning and control has drawbacks concerning monitoring and control, with respect to the many small disturbances that occur. Therefore, a novel architecture for production monitoring and control system based on the concept of intelligent products was presented. However, the feasibility of this approach has not been investigated. This chapter demonstrates this feasibility through experimental evaluation. For reasons of comparison, the TAC SCM simulation environment is used. The implementation of a TAC SCM manufacturer is presented, in which the intelligent products are aware of their local context and can negotiate with local manufacturing resources. Therefore, they can suggest local solutions to manufacturing problems virtually at the same time at which the problem occurs. This approach is compared with highly ranked TAC SCM manufacturer implementations. Besides financial results, robustness is used as an additional measurement of performance. The results of the simulations are encouraging.\footnote{This chapter appeared earlier as: G.G. Meyer and J.C. Wortmann. Robust planning and control using intelligent products. Agent-Mediated Electronic Commerce, pp. 163-177. Springer-Verlag, Lecture Notes in Business Information Processing 59, 2010, doi:10.1007/978-3-642-15117-0_12}
4.1 Introduction

Advances in production and supply chain planning and control over the past decades have steadily resulted in centralisation of the planning function. There are good reasons for this centralisation, both from a material perspective and from a capacity perspective. From materials perspective, coordination over the supply chain reduces the bullwhip effect [89, 114]. When combined with proper rules for safety stocks and lot sizes, this effect may almost be eliminated. Moreover, the problem of matched sets of parts in assembly requires coordination of supply streams for all components in the bill-of-material [137], which seems again to justify centralised planning. From capacity perspective, optimisation of one resource will usually impact other resources, such that some kind of coordination is not only useful but nearly unavoidable.

However, centralised planning and control also has its drawbacks, as for example is shown by [170]. These drawbacks appear in practise, and are caused by the many small disturbances that occur in manufacturing and transportation. A typical example of such a small disturbance is when a component is damaged, although it was planned to be used in manufacturing. In this case, a similar component needs to be sourced from somewhere else in order to continue with the original plan. Often, these kind of disturbances are not even made known to the central planners, as they are often solved on a more local level by for example a shop floor supervisor. Other kind of disturbances can include production errors and misshipments. These disturbances are one of the many causes why central plans in factories are seldom realised. Therefore, in the previous chapter, the architecture of a more robust monitoring and control system was proposed, based on the concept of intelligent products, which goal is to handle these disturbances in a more effective way.

In this chapter, the performance of the proposed system will be compared with other approaches, using the Trading Agent Competition Supply Chain Management (TAC SCM) simulated supply chain [34]. However, the usual measurement of performance in TAC SCM are the financial results, in terms of costs made and penalties paid balanced against profits made in
4.2 Background

Nowadays, there is an increasing interest in the field of intelligent products, and how intelligent products can be applied in different fields, such as in manufacturing and supply chain management (see Chapter 2). McFarlane et al. define an intelligent product as a physical and information-based representation of a product [116]. Figure 2.1 on page 21 shows an example of such a product. In this figure, the jar of spaghetti sauce is the physical product, the information-based representation of the product is stored in the database, and the intelligence is provided by the decision making agent. The connection between the physical product and the information-based representation is made using a tag and a reader, as will be further discussed later on. The fundamental idea behind an intelligent product according to Kärkkäinen et al. is the inside-out control of the supply chain deliverables during their lifecycle [98]. In other words, the product individuals in the supply chain themselves are in control of where they are going, and how they should be handled.

Recent technologies, such as automatic identification (Auto-ID), embedded processing, distributed information storage and processing, and agent based systems have been the main enablers for intelligent products. Auto-ID technologies, such as barcode and RFID, are commonly used to uniquely
identify individual products or delivery units. Especially RFID tags are suitable for tagging individual products, as multiple RFID tags can easily be read simultaneously, without requiring a line-of-sight, such as is the case with barcodes. In addition to automatic identification, Auto-ID technologies often also include localisation and sensor technologies. Localisation techniques, such as GPS, are often combined with automatic identification, as the location information is useless without the identity of the located entity [177]. Another frequently applied technique is updating the location status of the product at the moment its barcode or RFID-tag is scanned, when the physical location of the scanner is known [77].

The vision of intelligent products is to seamlessly connect the products in the physical world with their representation in information systems, e.g. through a product agent as proposed by [52]. Intelligent products would make it possible to avoid media breaks between the real world and the digital world. Thereby, data about the current and past state of products from the physical world can be retrieved and updated when needed. The basic building block for implementing a distributed information storage and processing system for products is that products are identified by globally unique identifiers that either encode links to information sources directly or that can be used as look-up keys in some kind of network infrastructure. The three main currently known approaches for distributed information storage and processing are EPCglobal[2][154], ID@URI[3][77], and WWAI[4]. A technical analysis and comparison of these approaches can be found in [50].

The agents paradigm is considered useful to implement the intelligence part of intelligent products. There are several reasons why the use of an agent-based platform for intelligent products is beneficial. Firstly, when there is a high number of products, the number of products in need of explicit control from the user has to be reduced. This can be achieved by making the products autonomous. In this way, intelligent products with knowledge and reasoning capabilities can do most of the repetitive tasks in an automated way. Secondly, intelligent products should be able to detect and react to changes in the environment. Agents can pro-actively assist the product and

---

try to achieve their goals in a changing environment. Agents can also help in discovering information about the environment by communicating with the agents of other products. It is therefore clear that intelligent agents have the characteristics which are desirable for intelligent products. Of course, an application for intelligent products can be created without the use of agents, but by using agents, one can take advantage of the methodologies and solutions provided by the multi-agent paradigm [27].

4.3 Methodology

To compare the performance of the proposed system design, as described in the previous chapter, with existing designs, the TAC SCM simulated supply chain is used [34], due to several reasons. Firstly, it was designed to capture many of the challenges involved in supporting dynamic supply chain practises, including challenges related to production monitoring and control. Further, it is a well-founded framework, and widely reported in literature (see e.g. [35, 57]). Finally, the framework can be easily extended and modified for specific needs.

Within a TAC SCM simulation, a maximum of six manufacturers of personal computers compete with each other for customer orders and for the procurement of a variety of components. For every otherwise identical computer manufacturer, a different production planning and control system can be deployed. In this way, the performance of different production planning and control systems can be compared. The TAC SCM scenario from the perspective of a single manufacturer can be seen in Figure 4.1. As shown in the figure, a manufacturer has four major tasks to perform, namely negotiate with suppliers for components, bid for customer orders, manage the production schedule and manage the shipping schedule. Further, each manufacturer has an identical assembly cell capable of assembling any type of computer, and a warehouse that stores both components and assembled computers.

In the current TAC SCM simulations and competitions, the performance indication of a manufacturer is solely based on the financial result, in terms of costs made for material, storage and penalties paid balanced against
profits made in sales. In principle, the manufacturer with the highest bank account at the end of a simulation run wins that run. This measurement of performance gives a good indication of which manufacturer is the most efficient one, in terms of costs and benefits. However, it does not provide a good indication about the robustness of the manufacturer, in case when the manufacturer has to deal with disturbances. For showing the robustness of a manufacturer, a measurement is needed which only indicates the capability of a manufacturer to handle unexpected disturbances in a flexible way. The financial results of the manufacturers give an indication of the overall performance, but robustness is only a minor part of that. Therefore, an additional measurement is used here. This measurement is the percentage of customer orders that are delivered to the final customer in time, i.e. if the delivery is before or on the due date of the specific order. This is considered to be a good measurement for the robustness of a manufacturer, as it gives an indication about the capabilities of a manufacturer to still deliver products to a customer in time, even when disturbances are happening.

Although there are some variations among the scenarios that manufacturers have to deal with, the standard TAC SCM scenario purposefully excludes disturbances. For the purpose of testing the performance of a manufacturer in terms of monitoring and control, a disturbance has been added to the simulated scenario. In the slightly modified version of the TAC SCM scenario, every component which is delivered by a supplier to a manufacturer
has an $n$ percent probability of being rejected. When this occurs, the component will not be added to the manufacturer’s inventory. This amounts to a material shortage disturbance, the most common disturbance in practise \cite{104}. In reality, such disturbances can have a variety of reasons, such as components being damaged, broken, delayed or wrongly shipped. With this additional disturbance added to the simulated scenario, experiments have been conducted with three different values for $n$, namely:

$n = 0$. In this case, none of the delivered components will be unusable. Therefore, this scenario is the same as the original TAC SCM scenario.

$n = 5$. In this case, every component has a chance of 5\% of being unusable.

$n = 10$. In this case, every component has a chance of 10\% of being unusable.

In order to achieve reasonable confidence in the results, the experiments were repeated 25 times for every value of $n$. Besides the proposed manufacturer implementation, the same competing manufacturer implementations were used in every experiment, namely: TacTex-07 \cite{138, 139}, PhantAgent-07 \cite{175}, DeepMaize-07 \cite{93} and Mertacor-08 \cite{29, 181}. These ‘opponents’ were chosen for their high rankings in recent TAC SCM competitions, as well as their availability on the agent repository of the TAC website\footnote{http://www.sics.se/tac}. The next section of this chapter describes how the proposed monitoring control system of Chapter 3 is implemented as a TAC SCM manufacturer. Following this, the simulation results are presented.

4.4 Prototype implementation

This section describes the system prototype, and how it is implemented as a TAC SCM manufacturer. The implemented manufacturer was named GRUNN within the conducted simulations. The GRUNN manufacturer can be downloaded from the agent repository on the TAC website, as well as
In this section, the description of the prototype is split into two parts, namely a part discussing the system structure, and a part discussing the system behaviour.

### 4.4.1 Structure

The basic idea of the implemented manufacturer system for TAC SCM simulations is illustrated in Figure 4.2, which shows a UML class diagram, in which the various internal agents of the GRUNN manufacturing are depicted. As shown in the figure, there are four different planner agents in the system, each to perform one of the four basic TAC SCM tasks as described earlier. In addition, the product agent has to perform tasks for monitoring and control of a single product, as described in Section 3.4.3, and is responsible for the successful production and delivery of this single product.

In case of the TAC SCM simulation, one order in the simulation is considered as one product in the presented system design. This however does not have any consequences in implementing the structural design presen-
Further, as the TAC SCM simulation does not allow for negotiation with human planners, the product agents will not use the decision-making mechanism described in Section 3.4.3 for proposing solutions, rather this mechanism will be used to create the overall production plan. As such, the responsibility of a product agent for completing an order covers the procurement of the components required for the assembly from the warehouse, the allocation of the required production capacity and arranging the shipment of the finished products to the customer.

The purpose of each agent type present in the system, including the planner and product agents, will be shortly described next.

- The **purchase planner agent** is responsible for acquiring components, which are required for the production of the to be delivered products. However, most of the tasks of this agent are transferred to other agents, as the purchase planner agent creates a separate agent for each component type. Such a separate component type agent is responsible for all the tasks related to one particular component type.

- The **sales planner agent** is responsible for acquiring orders. However, most of the tasks of this agent are transferred to other agents, as the sales planner agent creates a separate agent for each product type. Such a separate product type agent is responsible for all the tasks related to one particular product type.

- The **production planner agent** is responsible for assigning production capacity to products which are in need of assembly.

- The **shipment planner agent** is responsible for shipping assembled products to the waiting customers.

- A **component type agent** is responsible for acquiring components of one certain type. For this, every component type agent needs to negotiate with the suppliers of this component type. Furthermore, a component type agent is responsible for the distributing of this one type of components among unfinished products.

- A **product type agent** is responsible for acquiring orders of one certain
product type. For this, every product type agent needs to negotiate with potential customers.

- A *product agent* is responsible for the complete processing of one final product. In the case of TAC SCM, every customer order is considered to be a product, as every customer order can be seen as an individual and unique product which needs to be delivered by the manufacturer to the customer. Therefore, every customer order will have one product agent assigned to it, which makes the customer order an intelligent product. The responsibility of the product agent includes the procurement of components required for the assembly, the procurement of the required production capacity, as well as arranging the shipment of the finished products to the customer.

### 4.4.2 Behaviour

This subsection will describe the behaviour of the three most important agent types within the manufacturer implementation: the component type agent, the product type agent, and the product agent.

**Component type agent**

Every component type agent needs to acquire sufficient components of one certain type. For this, the behaviour of Figure 4.3 is applied by every component type agent. The figure shows a UML communication diagram, in which the communication of a component type agent with a supplier can be seen. This act of communication consists of three steps, which will be discussed next.

First, the component type agent will send Request For Quotes (RFQs) to every supplier, which can deliver the component type this agent is respons-
4.4. PROTOTYPE IMPLEMENTATION

Figure 4.4: Behaviour of a product type agent

ible for. The amount of components as well as the delivery date asked for in an RFQ are based on sales estimations, the quantity that is still in inventory, and the quantity that is ordered but still needs to be delivered. This sales estimation is based on (historical) information which the component type agent receives from the different product type agents. Secondly, suppliers will send quotes back to the component type agent, telling the agent how much they can deliver, on what date, and for what price. Finally, the component type agent will compare the different quotes, and respond by sending orders back to the suppliers who had the best quotes for this component type. Which quote is considered to be the best quote is primarily based on the price per component, but when prices are almost the same it is also based on the quantity and the delivery date.

Product type agent

Every product type agent needs to acquire orders for products of one certain type. For this, the behaviour of Figure 4.4 is applied by every product type agent. The figure shows a UML communication diagram, in which the communication of a product type agent with a customer and a product agent can be seen. This act of communication consists of four steps, which will be discussed next.

First, the product type agent will receive RFQs of customers, in case customers are requesting quotes for products of the type this agent is responsible for. Each RFQ will contain information about the amount of products, as well as a due date. Secondly, the product type agent will respond with a quote, when the agent considers it feasible to deliver the product before the due date of the customer with a positive financial result. To achieve this, the agent will calculate a price per product based on an estimation of the current market price and adjusted according to the current factory load.
This price is compared with the costs of the required components, resulting in a decision whether the quote will be sent to the customer or not. Thirdly, when a customer considers the quote of the product type agent the best compared to the other manufacturers, the customer will send back an order. Finally, for every customer order the product type agent receives, a product agent is created, which will be responsible for the complete processing of this one order.

Product agent

As mentioned before, a product agent is responsible for the complete handling and processing of one particular order. For this, the behaviour of Figure 4.5 is applied by every product agent. The figure shows UML communication diagrams, in which the communication of a product agent with a component type agent, a production planner agent, and a shipment planner agent can be seen. These communication acts are part of the different planning tasks in which the product agent is playing a role. These different planning tasks in which the product agent is involved will be discussed in more detail next.

The component planning is the first planning task in which the product
agent is involved. Product agents should be able to assist the component type agent in distributing available components among the different products who require components for production. This functionality requires the intelligent product to already exist before the actual product is produced, i.e. the intelligent product is already in existence from the moment that there is the intention to make the product. This distribution of components among products should be based on priority, therefore, products with earlier due dates should get priority above products with later due dates. In order to achieve a distribution of components based on priorities, an auction based negotiation system is used, which consists of several steps. First, every component type agent will send a Request For Bids to all product agents, when it has components to distribute. Secondly, every product agent who is in need of this component type will send a bid to this component type agent, containing the amount of components of this type it needs, as well as the offered price per component. In this approach, the price per component the product agent is offering will increase when the amount of days left till the due date of the specific order is decreasing. Finally, the component type agent will inform all agents who have send a bid whether they have won the components or not. The product agents with the highest bids will always win the auction, as long as the component type agent has enough components in stock.

The production planning is the second planning task in which the product agent is involved. Product agents should be able to assist the production planner agent in distributing the available production capacity among the different products who require production. As with the component planning, the distribution of production capacity among products should be based on priority, therefore, products with earlier due dates should get priority above products with later due dates. In order to achieve a distribution of production capacity based on priorities, an auction based negotiation system is used, which consists of several steps. First, the production planner agent will send a Request For Bids to all product agents, when it has production ca-
capacity to distribute. Secondly, every product agent who is in need of production will send a bid to the production planner agent, containing the amount of production capacity it needs, as well as the offered price per production unit. In this approach, the price per production unit the product agent is offering will increase when the amount of days left till the due date of the specific order is decreasing. Finally, the product planner agent will inform all agents who have send a bid whether they have won the production capacity or not. The product agents with the highest bids will always win the auction, as long as the production planner agent has enough production capacity available.

The shipment planning is the third planning task in which the product agent is involved. Product agents should be able to assist the shipment planner agent in planning the shipments of finished products to the customers. Differently than the component planning and production planning, no prioritising is needed, as there is no limitation on the shipment capacity in case of the TAC SCM scenario. However, for design consistency, the applied approach assumes a limited shipment capacity, which therefore requires prioritisation. In order to achieve a distribution of shipment capacity based on priorities, an auction based negotiation system is used, which consists of several steps. First, the shipment planner agent will send a Request For Bids to all product agents. Secondly, every product agent who is in need of shipment will send a bid to the shipment planner agent, containing the amount of shipment capacity it needs, as well as the offered price per shipment unit. Finally, the shipment planner agent will inform all agents who have send a bid whether they have won the shipment capacity or not. However, in case of the TAC SCM scenario, there is no limitation on the shipment capacity available. Therefore, product agents with bids will always win the auction and will always get shipped.

The developed system will not result in the best possible plan because a centralised system is always able to find a more-optimal solution within a mathematical domain. Distributed systems are typically greedy and therefore suboptimal. However, as will be illustrated by the results in the next
4.5 Simulation results

This section presents the results from the simulation experiments, as described above. As described in the methodology section, three different experimental setups have been used, namely with zero, five, and ten percent of the delivered components being unusable, and therefore not delivered to the inventory of the manufacturer. The results presented in this section are based on the averages of the conducted simulations. For the GRUNN manufacturer, the standard deviations are also shown in every graph by means of error bars. The dummy manufacturer is omitted in the results presented in this section, as this manufacturer did not provide any relevant results. However, all detailed results including standard deviations for all manufacturers can be found in Appendix A on page 177.

The newly developed monitoring and control system did perform well when considering the robustness performance measure. This robustness measure is defined as the percentage of orders that are delivered to the final customer on time, i.e. the delivery of a specific order is on or before the due date. Figure 4.6 shows the results from the conducted simulations in terms of the percentage of orders finished in time versus the percentage of unusable components.
of orders finished on time. The graph shows that the percentage of orders finished in time is decreasing for all manufacturers when the percentage of unusable components is increasing. Only GRUNN is an exception to this. Even in the case where ten percent of all components are unusable, GRUNN still manages to finish nearly all orders in time. This observation confirms that an approach based on intelligent products can be very effective in handling disturbances in the simulated scenario.

Figure 4.7 shows the results of the conducted experiments in terms of profit. Two important observations can be made from the graph. Firstly, the graph clearly shows that for all three different experimental setups GRUNN does not perform as well as the other manufacturers in terms of profit. This observation is in line with our expectations. Secondly, for all manufacturers, the profit is decreasing when the amount of unusable components is increasing. This observation is also in line with our expectations, as manufacturers need to buy more components to finish the same amount of orders, when the amount of unusable components is increasing.

One obvious approach to overcoming the problem of unusable components is to increase the component inventory "safety stock" margin. Figure 4.8 shows the average storage costs per accepted order for each applied manufacturer system, and this gives a good indication of the inventory levels of
4.6 Conclusions

In this chapter, the following has been concluded:

- The TAC SCM simulated supply chain is very suitable for demonstrating the performance of production planning and control systems.

- A robustness measure and an additional disturbance have been added to the TAC SCM scenario, in order to test the performance of manufacturers in terms of monitoring and control.

- A prototype implementation of the production monitoring and control system based on the concept of intelligent products is presented.

- Experimental evaluation with the TAC SCM simulated scenario has shown that intelligent products perform very well in terms of robustness, but poor in terms of profit.

Figure 4.8: Storage costs of manufacturers per accepted order

each manufacturer. The figure clearly shows that using the GRUNN approach does not lead to a significantly larger inventory, and therefore that it is not dealing with the problem of unusable components by increasing safety stock levels.
The intelligent products approach showed to be very promising for monitoring and control purposes, when robustness is considered as an important factor.