



**HAL**  
open science

# The application of distributed intelligence to reduce the bullwhip effect

David de La Fuente, Jesus Lozano

► **To cite this version:**

David de La Fuente, Jesus Lozano. The application of distributed intelligence to reduce the bullwhip effect. *International Journal of Production Research*, 2007, 45 (08), pp.1815-1833. 10.1080/00207540500456021 . hal-00512878

**HAL Id: hal-00512878**

**<https://hal.science/hal-00512878>**

Submitted on 1 Sep 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



**The application of distributed intelligence to reduce the bullwhip effect**

Journal:	<i>International Journal of Production Research</i>
Manuscript ID:	TPRS-2005-IJPR-0186.R1
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	13-Oct-2005
Complete List of Authors:	de la Fuente, David; Oviedo University, Business Administration and Accountancy Lozano, Jesus; Oviedo University, Business Administration and Accountancy
Keywords:	FORECASTING, AGENT BASED SYSTEMS
Keywords (user):	



# The application of distributed intelligence to reduce the bullwhip effect

David de la Fuente (\*); Jesús Lozano

E.P.S.I.G. Department for Business Administration and Accountancy, the University of

Oviedo

(\* ) david@uniovi.es

## OVERVIEW

This article applies distributed intelligence to the bullwhip or Forrester effect, which it manages to reduce in a range of time series to which a genetic algorithm was applied. There are four parts to the article. The first provides an overview of the Forrester, or Bullwhip, effect. The second not only describes the genetic algorithms in terms of being devices that provide the model with intelligence but also introduces the agent network and the general model that supports them. The third describes the software used in the model that this article describes. The final section provides numeric examples, and draws a number of conclusions.

*Keywords:* Bullwhip, Agents, Genetic Algorithms, Forecast

## 1. THE BACKGROUND OF THE BULLWHIP EFFECT.

The study of the Forrester effect has been a productive field of research in logistics. The first formal description of the effect is attributed to Forrester himself in 1961. This was then fleshed out by Sterman (1989, 1992, 1995), who included the famous example of the beer distribution game that is played out in many a business school. According to Sterman, the Forrester effect stems from a non-optimal solution applied by members of the supply chain who see their strategies as a sum of individual strategies rather than a unit. Kaminsky and Simchi-Levi (1998) developed different IT versions of the beer distribution game, highlighting three intertwined effects on the supply chain: the Forester effect, the centralisation effect and the supply time effect. A range of financial studies on the long-term bullwhip effect highlight the reasons why the effect occurs when rational decisions are taken to try to mitigate the uncertainty of demand (Blanchard (1983), Blinder (1986), West (1986), Kahn (1987)).

In 1997, Lee et al. studied the problem by applying analytical models. They concluded that there were four basic reasons why the Forrester effect occurs: forecasting errors, fear of lack of products, inadequate lot sizing and price fluctuations. More recently, Cachon and Lariviere (1999) studied the effect of scarcity; Kille y Milne (1999) and Cachon (1999) studied the effect of lot sizing; Drezner et al. (1996), Baganha and Cohen (1998), Graves (1999), Chen et al. (2000), Aviv (2001, 2003) and Dejonckheere et al (2003) studied demand-related error. However, Disney and Towill (2001, 2002, 2003a, 2003b, 2004a, 2004b, 2004c) are the most prolific authors on the subject. The former wrote his doctoral thesis (2001) on the methodology of Virtual Managed Inventory (VMI) and then widened the scope of his studies to consider the bullwhip effect from the point of view of control theory. Other researchers, such as Disney y Grubbstrom (2004), Chiderhouse et al. (2004), Dejonckheere et al. (2004), Disney et al. (2004)) have also published on the same issue.

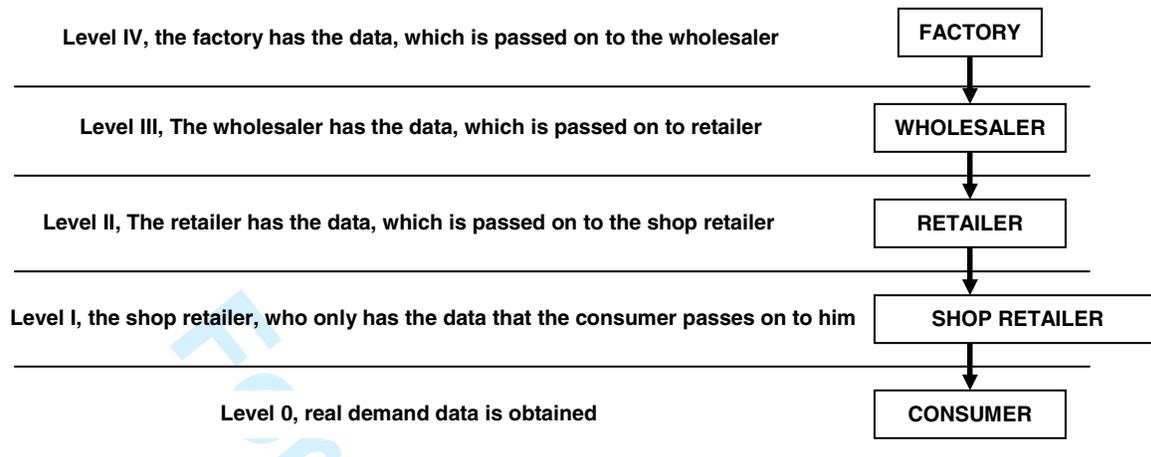


Figure 1: four supply chain scenarios.

Figure 1 describes the typical structure of distribution logistics. Rather than information flowing upstream across all the levels, it only flows between two consecutive ones (Sterman 1989). A number of cases in which there is information feedback from sales outlets to all levels of the supply chain have been studied in simulation cases (c.f. Mason-Jones 1998) others with control theory and yet others with z transformation (Dejonckheere et. al., 2003).

## 2. MODELS TO CURB THE FORRESTER EFFECT. GENETIC ALGORITHMS.

Figure 2 describes the typical cycle of a Genetic Algorithm, which evolves as follows: assuming an initial population of  $N$  possible solutions for the problem, where  $N$  is the size of the population:

$$X^k = (X_{11}^k, \dots, X_{1L}^k, X_{21}^k, \dots, X_{2L}^k, \dots, X_{M1}^k, \dots, X_{ML}^k) \quad k = 1, \dots, n$$

where  $k$  is the level of the supply chain. In our particular case,  $N$  is four. The index, which ranges from 1 to  $M$ , represents the different characteristics to be measured at a given level, such as stock, orders, and so forth. The index in which  $L$  varies represents the time periods. Generally speaking, the set of genes (elements), for each chromosome (individual) is selected as follows:

- The first gene is randomly selected.
- the remaining genes are also then randomly selected, but in such a way as to comply with the restrictions imposed by demand. They are checked to make sure that they meet stop conditions.

Population  $H$  is then selected, with only  $N \cdot G$  elements, where  $G$  is the crossover index. The  $N \cdot G$  elements of the population that have the highest value for  $f$  (fitness function) will be selected. Crossover and mutation operations occur during the reproduction stage. Crossover is linear, and thus two parents ( $p_1$  y  $p_2$ ) of size  $N'$  are randomly selected and generate two offspring. The child with the best "fitness" is selected.

Mutation (the frequency of which depends on parameter M) consists of interchanging two randomly selected components of a child that has also been randomly selected. The reproduction phase occurs as many times as is necessary to obtain  $(1-G)*N$  children.

We thus have Genetic Algorithms at each level, with binary individuals (0,1) modified using the fitness by roulette-wheel sampling method, crossover index of 0,9, and a mutation index of 0,01 and 500 individuals.

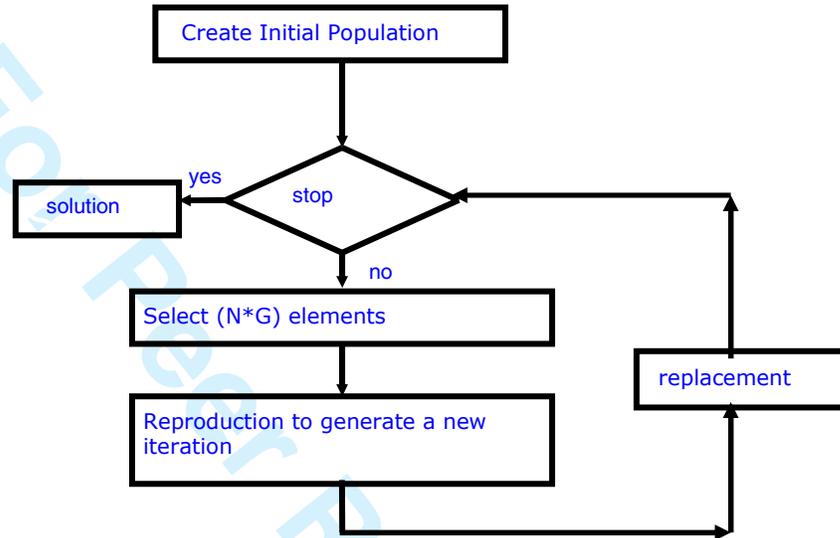


Figure 2. A Genetic Algorithm loop

The pseudo-code of the fitness functions for each level is described next (see appendix):

#### SHOP RETAILER

```

sub fitdeta1 {
  w = .5+rand(0:1);
  media=0;
  for z (p-1..período actual p){ # loop from z=p-1 up to z=p
    x=demanda[z]-invfdeta1[z]+rstockdeta1[z];
    y = abs(w*x); # absolute value
    media += y; # acumulated
  }
  deta1=media/2; # which is ordered to the retailer (this data is exported)
  return 1/deta1; # the minor, the better, the inverse is taken
}
  
```

#### RETAILER

```

sub fitmino {
  w = .5+rand(0:1);
  media=0;
  for z (p-1..p){
    x=pidedeta1[z]-invfmino[z]+rstockmino[z];
    y = abs(w*(x+deta1)/2);
    media += y;
  }
  mino=media/2;
  return 1/mino;
}
  
```

```

1
2
3 }
4
5 WHOLESALE
6 sub fitmayo {
7   w = .5+rand(0:1);
8   media=0;
9   for z (p-1..p){
10    x=pidemino[z]-invfmayo[z]+rstockmayo[z];
11    y = abs(w*(x+deta1)/2);
12    media += y;
13  }
14  mayor=media/2;
15  return 1/mayor;
16 }
17
18
19 FACTORY
20 sub fitfabr {
21   w = .5+rand(0:1);
22   media=0;
23   for z (p-1..p){
24    x=pidemayo[z]-invffabr[z]+rstockfabr[z];
25    y = abs(w*(x+deta1)/2);
26    media += y;
27  }
28  proc=media/2;
29  return 1/proc;
30 }
31

```

Table 1 shows how initial data is incorporated for the Genetic Algorithm that we are proposing. Results will be fleshed out in the final section, which describes numeric applications. The data comes from Sterman's beer distribution example (1989).

PERIOD 1	Per. 2	Per. 3	Per. 4	PERIOD 5
Demand of 1 = 95	80	80	85	Demand of 10 = 90
Initial Inventory. Shop retailer 1 = 100	95	80	65	Initial Inventory. Shop retailer. 10 = 40
Shop retailer supplies 1 = 95	80	80	65	shop retailer supplies 10 = 40
Stockout 1 =			20	Stockout 10 = 50
Final inventory Shop retailer 1 = 5	15	0	0	Final inv. Shop retailer 10 = 0
Shop retailer orders 1 = 90	65	80	85	Shop retailer orders 10 = 90
Initial inventory, retailer 1 = 100	90	65	40	Initial Inv. retailer 10 = 60
retailer supplies 1 = 90	65	65	40	retailer supplies 10 = 60
Stockout 1 =		15	45	Stockout 10 = 30
Final inventory retailer 1 = 10	25	0	0	Final inventory retailer 10 = 0
Retailer orders 1 = 80	40	80	85	Retailer orders 10 = 90
Initial inventory wholesaler 1 = 100	80	40	60	Initial inventory wholesaler 10 = 80
Wholesaler supplies 1 = 80	40	40	60	Wholesaler supplies 10 = 80
Stockout 1 =		40	25	Stockout 10 = 10
Final inventory wholesaler 1 = 20	40	0	0	Final inventory wholesaler 10 = 0
Wholesaler orders 1 = 60	0	80	85	Wholesaler orders 10 = 90

Initial inventory manufacturer 1 = 100	60	60	80	Initial inventory manufacturer 10 = 85
Factory supplies 1 = 60	0	60	80	Factory supplies 10 = 85
Stockout 1 =		20	5	Stockout 10 = 5
Final inventory factory 1 = 40	60	0	0	Final inventory factory 10 = 0
Factory manufactures 1 = 20	0	80	85	Factory manufactures 10 = 90

Table 1

Table 2 provides the solution using the so-called 1-1 algorithm (Sterman 1989) and the Genetic Algorithm described here. In both cases, results are obtained from 5 data. It is clear that for this particular case the solutions provided by both algorithms coincide in having high error levels of variance manufacturing.

FINAL INVENTORIES, 1-1 CASE	FINAL INVENTORIES, GA CASE
Average demand 86	Average demand 86
Maximum 95 minimum 80	Maximum 95 minimum 80
Demand variance 34	Demand variance 34
Average shop retailer 4	Average shop retailer 6
Maximum 15 minimum 0	Maximum 25 minimum 0
Variance shop retailer 34	Variance shop retailer 94
Average, retailer 7	Average, retailer 8.8
Maximum 25 minimum 0	Maximum 44 minimum 0
Variance retailer 96	Variance retailer 309.76
Average wholesaler 12	Average wholesaler 25
Maximum 40 minimum 0	Maximum 85 minimum 0
Variance wholesaler 256	Variance wholesaler 1140
Average factory 20	Average factory 39.8
Maximum 60 minimum 0	Maximum 75 minimum 0
Variance factory 640	Variance factory 879.76
Average manufacturing 55	Average manufacturing 113.4
Maximum 90 minimum 0	Maximum 199 minimum 20
Variance manufacturing 1400	Variance manufacturing 5195.44
TOTAL STOCKOUT 95	TOTAL STOCKOUT 31

Table 2

Table 2 is the starting point for the agent to decide how to plan deliveries, what to hold in stock, and so forth. The different members of the supply chain also have this information, and, as each knows the real costs, they will choose the best option for them (costs have not been used to resolve the Genetic Algorithm, as can be seen in the fitness functions)

Figure 3 provides an overview of the multi-agents for our supply chain. The concept of 'agent' as used in this text includes applying a genetic algorithm to decide how much to order in order to have stock. However, it also includes using other information provided by such tools (which will be described later) as neural networks, Box-Jenkins and other forecasting methods as well as simulation not only to validate orders sent and received but also to obtain rules. Furthermore, by applying the Corba tool (which will also be commented upon at a later point in the text), the Genetic Algorithm can be

solved using different platforms; as agents can communicate with KQML, agreement can be reached on how much to dispatch, order, and so on.

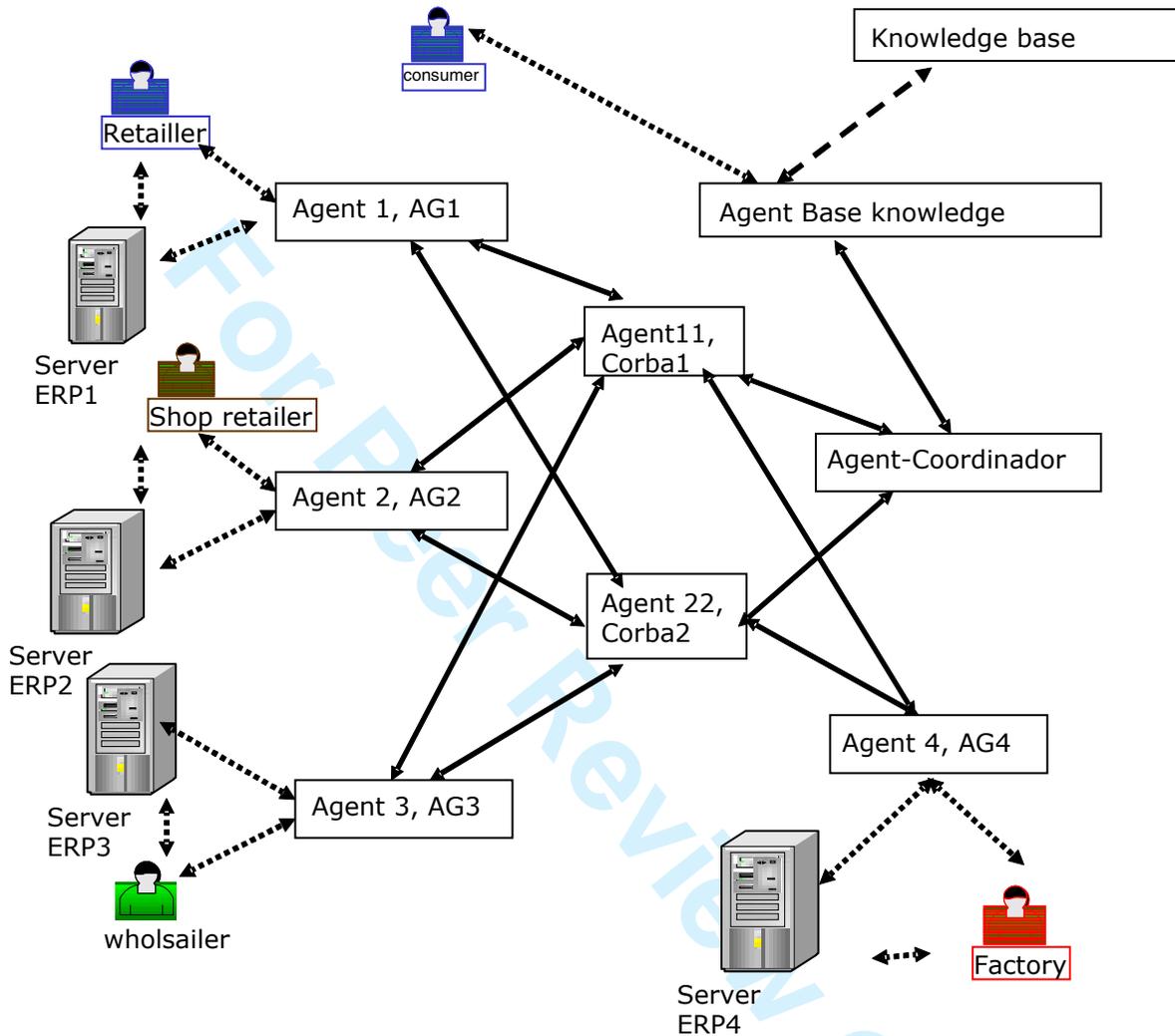


Figure 3. A four-level network

It should be pointed out that the mediocre results of Table 2 are very much improved upon in other runs or simply when we move up to 20 data. This is even more the case if we use more than 60, as advanced forecasting methods such as Neural Networks and Box-Jenkins come into play. Using simple forecasting methods with 100 number demand series has demonstrated to us that the so-called simple techniques (constant, moving average, exponential smoothing, trends or random walk, etc.) worked worse than advanced methods, i.e., Neural Networks and Box-Jenkins, as is only to be expected.

The second facet to be considered is for each participant in the supply chain to have the same inventory and for orders to be made in line with the following expression:

$$Y_{k+1,t} = Y_{k,t} + (S_{k,t} - S_{k,t-1})$$

where  $S_{k,t}$  is the stock required to be level with past demand for a supply chain level  $k$ . This expression is linked to the production planning equations of each independent level.  $X_{k,t}$  represents what will be dispatched to the client.

$$S_{k,t-1} + X_{k,t} - S_{k,t} = Y_{k,t}$$

In the case in question in this paper, there is a single sub-index,  $k$ , since there is a single product; if there were  $i$  products, we would express the general formula as:

$$Y_{i,k,t}$$

Recently, many researchers have used the following equation to measure the bullwhip effect, in which ORATE represents orders of all those offering products (retailers, factory etc). As a general rule, we will use the factory, or manufacturer, as this is most distant from the consumer and its coefficient is greater. CONS represents sales to the end consumer.

$$RV_{ORATE} = \frac{\sigma^2_{ORATE} / \mu_{ORATE}}{\sigma^2_{CONS} / \mu_{CONS}} = \frac{\sigma^2_{ORATE}}{\sigma^2_{CONS}}$$

Fluctuations in the real inventory are easily measured using the following variation ratio,  $RV$ ,

$$RV_{INVR} = \frac{\sigma^2_{INVR} / \mu_{INVR}}{\sigma^2_{CONS} / \mu_{CONS}} = \frac{\sigma^2_{INVR}}{\sigma^2_{CONS}}$$

Although any other software could be used, we used Perl language with the following characteristics: it works with procedural as well as object-oriented programming, and the Perl interpreter can run on other systems. The actual version was the 5.8.6. Genetic Algorithm Sub-routines using Linux. Corba architecture comes from the Object Management Group Inc. (<http://www.omg.org>), founded in 1989 to develop reutilisation, portability and interoperativity of software objects in heterogeneous and networked environments. The consortium created an architecture called Object Management Architecture (OMA), which Corba forms part of. The architecture was constructed so that it could run on any platform.

### 3. NUMERIC RESULTS.

We divided the application according to the amount of order data that was available, i.e., into when we have scant data of between 10 and 20, and when we have around 100.

#### 3.1. Demand of between 10 and 20 data.

We will first describe the applications using Perl and Corba after designing the Genetic Algorithm described in previous sections of the text. This is followed by a numeric and graphic presentation of the tests that were run. The hardware used was a CPU AMD Duron 800 Mhz 512 Mb RAM PC, and the software was as follows: operative system: Linux Debian; Perl v5.8.4 (with AI (Artificial Intelligence) packets: Genetic, Bundle::Math::Random); gnuplot v4.0, and the Perl Genetic Algorithm prototype.

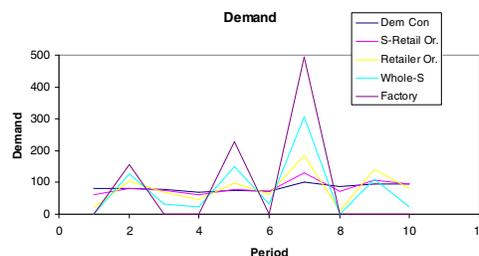
The improvement in variance when Genetic Algorithms are applied is noteworthy (c.f. Table 3), although this method tends to be rather slow, requiring more than 30 minutes on an 800Mhz AMD Duron for a 20 period resolution. Rather than being constant, time elapsed can vary wildly for another resolution under the same conditions. Resolution time for the 1-1 Model is negligible, as would be expected.

The columns of Table 3 provide the following values from left to right: the first column is of the different time series; column two is the number of cases used in each of these time series. Column three elucidated which of the two models – the 1-1 Model or the Genetic Algorithm–applies; column 4 shows variances of factory demand and of the end consumer. The fifth column is time taken to find a solution using a standard computer. The final column is the measurement of the Bullwhip effect,  $RV_{ORATE}$ .

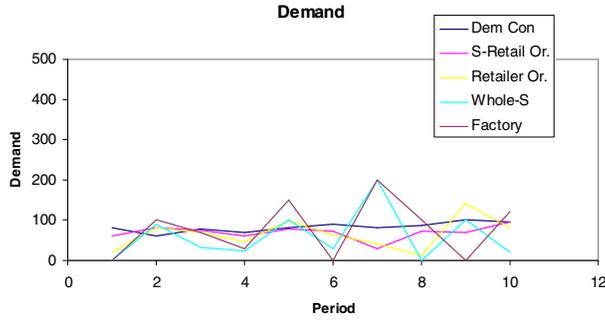
Series	Number of cases	Model	$\frac{\sigma^2_{ORATE}}{\sigma^2_{CONS}}$	Time	$RV_{ORATE} = \frac{\sigma^2_{ORATE}}{\sigma^2_{CONS}}$
<i>N(85,10)</i> <i>Test 1</i>	10	<i>Model 1-1</i>	<i>17.652,16</i>	<i>1 ms</i>	28,9
		<i>GA1</i>	<i>609,89</i>	<i>25 minutes</i>	
N(85,10) Test 2	15	1-1 Model	13.536,16	1 ms	5,3
		GA1	2.536,81	20 minutes	
N(85,10) Test 3	25	1-1 Model	16.175,61	1 ms	2,5
		GA1	6.410,81	22 minutes	
N(85,10) Test 4	10	1-1 Model	21.716,64	1 ms	11,2
		GA1	1.930,06	18 minutes	
N(85,5) Test 5	20	1-1 Model	7.909,84	1 ms	7,01
		GA1	1.127,44	8 minutes	
N(85,1) Test 6	20	1-1 Model	2.223,49	1 ms	2,23
		GA1	996,09	5 minutes	
<i>N(85,0.5)</i> <i>Test 7</i>	10	<i>1-1 Model</i>	<i>1.805,98</i>	<i>1 ms</i>	1,82
		<i>GA1</i>	<i>987,59</i>	<i>2 minutes</i>	
N(85,15) Test 8	20	1-1 Model	32.648,21	1 ms	12,49
		GA1	2.613,84	40 minutes	

Table 3

As can be seen, the results described in Table 3 clearly highlight the efficacy of the Genetic Algorithm compared to the 1-1 model in every single case. Furthermore, when  $\sigma$  is low (e.g. 0.5) the variance of the two models is similarly low, although always higher than the non-intelligent model. The measure of the bullwhip effect stands at 28,9 for Test 1 and at 1,82 for Test 7. The graphs that follow show change in demand with the two methods, test 1 and 7 in italic. Variance is far greater in Test 1. However, the smoothing achieved by the Genetic Algorithm is spectacular in graphs 1 and 2, since at some points it drops to productions of near to 0 units when real demand is around 100, the GA does not exceed 110 units.



Graph 1. All-level demand using the 1-1 Method, with N(85,10)

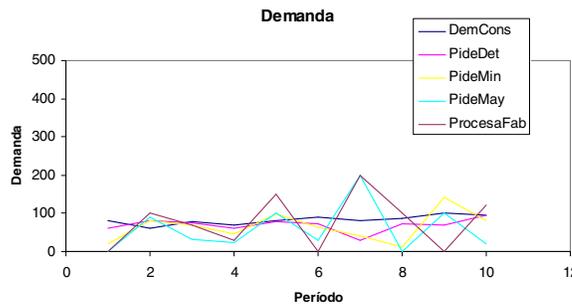


Graph 2. All-level demand using the GA, with N(85,10)

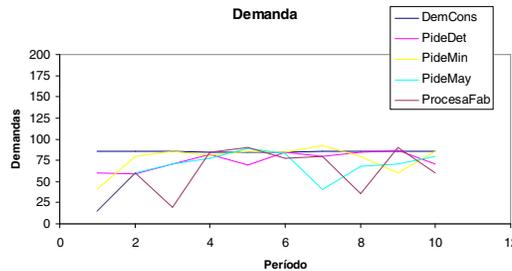
Table 4 gives the results of both the 1-1 Model and the GA methods (The GA is superior, as in most calculations)

Model 1-1	Genetic Algorithm
Factory average 20.9	Factory average 21
Maximum 100 minimum 0	Maximum 90 minimum 0
Factory Variance 1028.89	Factory Variance 906
Factory initial inventory 10 = 96	
Factory supplies 10 = 95	
Stock out 10 =	
Factory final inventory 10 = 1	
Factory manufactures 10 = 94	

Table 4



Graph 3. All-level demand using the 1-1 Method, with N(85,0.5)



Graph 4. All-level demand using the GA, with N(85,0.5)

### 3.2. Real demand size (over 100 data) with previsions forecasts using advanced methods

Forecasts were made for 19 time series obtained from databases that are generally used in forecasting models, such as Abraham (1983), Box-Jenkins (1976) and Reilly (1980). Table 5 indicates whether they are stationary and whether they are seasonal, i.e., four patterns adapted from Zhao et al. (2002) and labelled as follows: ESES, NESES, ESNES y NESNES. Table 6 shows the name, origin of the series, content, number of data and number of forecasts.

Yes stationary	Yes seasonal	ESES (4)
No stationary	Yes seasonal	NESES (6)
Yes stationary	No seasonal	ESNES(1)
No stationary	No seasonal	NESNES (8)

Table 5

SERIES	SOURCE	RELATING TO	Nº data	Nº forecasts
<b>AL03</b>	Abraham (1983)	Electricity consumption	106	12
<b>AL04</b>	Abraham (1983)	Car sales	108	13
<b>AL09</b>	Abraham (1983)	Mortgage - Loan differences	159	9
<b>AL011</b>	Abraham (1983)	Gas consumption	106	11
<b>AL12</b>	Abraham (1983)	Demand for petrol	192	12
<b>BJ02</b>	Box y Jenkins (1976)	Price of IBM shares	369	9
<b>BJ06</b>	Box y Jenkins (1976)	Wolfer sunspots	100	11
<b>BJ08</b>	Box y Jenkins (1976)	Airline company passengers	144	10
<b>BJ15</b>	Box y Jenkins (1976)	Warehouse sales	150	12
<b>UN01</b>	Reilly (1980)	Simulated	150	10
<b>UN02</b>	Reilly (1980)	Simulated	162	10
<b>UN03</b>	Reilly (1980)	Simulated	150	10
<b>UN04</b>	Reilly (1980)	Simulated	161	10
<b>UN05</b>	Reilly (1980)	Simulated	155	10
<b>UN06</b>	Reilly (1980)	Simulated	178	10
<b>UN07</b>	Reilly (1980)	Simulated	149	10
<b>UN08</b>	Reilly (1980)	Simulated	148	10
<b>UN17</b>	Reilly (1980)	Simulated	178	10
<b>UN21</b>	Reilly (1980)	Simulated	146	10

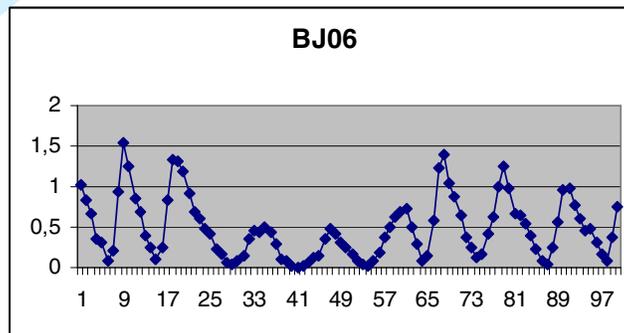
Table 6.

#### FORECASTS USING BOX-JENKINS AND ARTIFICIAL NEURAL NETWORKS.

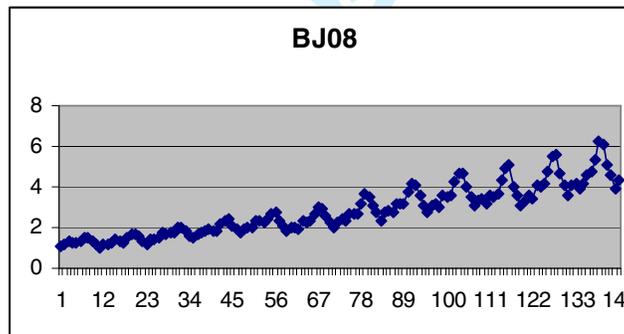
Table 7 indicates the type of series in terms of whether it is stationary, seasonal, and so forth. Results using the Box-Jenkins methodology were obtained with Statgraphics 5.1. and SCA programmes and are documented in table 8. Graphs 5 and 6 describe the

BJ06 and BJ08 data series as an example. The latter of the series 10% of the data was used to validate the models, i.e., if the series contained 100 data, 90 of them were used for fitting, both for Box-Jenkins and Networks, and the remaining 10 served for forecasting and for comparison with real data. The NeuroPlanet programme was used for Networks.

A single hidden layer was seen to be sufficient to solve the problem in the case of Neural Networks, and the architecture described in table 8 is of the "d-n-s" type, where d is the number of input layers, n is the neurons in the hidden layer and s is the neurons of the output layer, whose value is 1 since we are interested in step by step forecasting. The remaining parameters, that is "d" and "n" are obtained by testing different values until one is found that cannot be improved upon. Table 10 describes the values we obtained. Forecasting results for both methods were normalised to values between 90 y 110 so that they could be compared with the 1-1 model.



Graph 5



Graph 6

SERIES	Stationary	seasonal	TYPE
AL03	YES	YES	ESES
AL04	NO	YES	NESES
AL09	NO	NO	NESNES
AL011	YES	YES	ESES
AL12	NO	YES	NESES
BJ02	NO	NO	NESNES
BJ06	NO	NO	NESNES
BJ08	NO	YES	NESES
BJ15	NO	NO	NESNES
UN01	NO	YES	NESES

UN02	YES	YES	ESES
UN03	NO	NO	NESNES
UN04	YES	YES	ESES
UN05	NO	YES	NESES
UN06	NO	YES	NESES
UN07	NO	NO	NESNES
UN08	YES	NO	ESNES
UN17	NO	NO	NESNES
UN21	NO	NO	NESNES

Table 7. Model types

SERIES	BJ UNIVARIANT MODEL	BJ UNIVARIANT MODEL	ARCHITECTURE AND CONFIGURATION, NN	
AL03	$(1,0,0)(1,1,1)_{12}$	$(1-\phi_1B)(1-\phi_{12}B^{12})(1-B^{12})$ $x = (1-\theta_{12}B^{12})a_t$	MLP	6-3-1
AL04	$(1,0,0)(0,1,1)_{12}$	$(1-\phi_1B)(1-B^{12}) x =$ $k+(1-\theta_{12}B^{12})a_t$	MLP	12-5-1
AL09	$(1,0,0)$	$(1-\phi_1B) x = k+a_t$	MLP	5-3-1
AL011	$(1,0,0)(0,1,1)_{12}$	$(1-\phi_1B)(1-B^{12}) x =$ $k+(1-\theta_{12}B^{12})a_t$	ELM	6-3-1
AL12	$(0,0,3)(0,1,0)_{12}$	$(1-B^{12}) x = k+(1-\theta_2B^2-$ $\theta_3B^3)(1-\theta_{12}B^{12})a_t$	MLP	12-4-1
BJ02	$(1,0,0)$	$(1-\phi_1B) x = a_t$	MLP	7-4-1
BJ06	$(2,0,1)$	$(1-\phi_1B-\phi_2B^2) x = k+(1-$ $\theta_1B)a_t$	MLP	6-2-1
BJ08	$(1,1,0)(0,1,0)_{12}$	$(1-\phi_1B)(1-B)(1-B^{12}) x =$ $a_t$	MLP	12-7-1
BJ15	$(1,1,1)$	$(1-\phi_1B)(1-B) x = (1-$ $\theta_1B)a_t$	MLP	8-4-1
UN01	$(0,1,1)(1,1,1)_{12}$	$(1-\phi_{12}B^{12})(1-B)(1-B^{12}) x =$ $(1-\theta_1B)(1-\theta_{12}B^{12})a_t$	MLP	12-7-1
UN02	$(1,0,0)(0,0,1)_{12}$	$(1-\phi_1B)(1-\theta_{12}B^{12}) x = a_t$	MLP	8-3-1
UN03	$(1,0,0)$	$(1-\phi_1B) x = k+a_t$	ELM	10-8-1
UN04	$(0,0,1)(0,1,0)_{12}$	$(1-B^{12}) x = (1-\theta_1B)a_t$	MLP	12-5-1
UN05	$(1,0,1)(0,0,1)_{12}$	$(1-\phi_1B)(1-\theta_{12}B^{12}) x =$ $k+(1-\theta_1B) a_t$	MLP	12-4-1
UN06	$(0,1,0)(0,1,0)_4$	$(1-B)(1-B^4) x = (1-$ $\theta_1B)a_t$	ELM	16-6-1
UN07	$(1,0,0)(1,0,0)_6$	$(1-\phi_1B)(1-\phi_6B^6) x = a_t$	MLP	12-6-1
UN08	$(2,0,0)(1,0,0)_6$	$(1-\phi_1B-\phi_2B^2)(1-\phi_6B^6) x =$ $k+a_t$	ELM	8-4-1
UN17	$(1,3,0)$	$(1-\phi_1B)(1-B)^3 x = a_t$	MLP	5-2-1
UN21	$(3,0,2)$	$(1-\phi_1B-\phi_2B^2-\phi_3B^3) x =$ $k+(1-\theta_1B + \theta_1B^2) a_t$	MLP	6-3-1

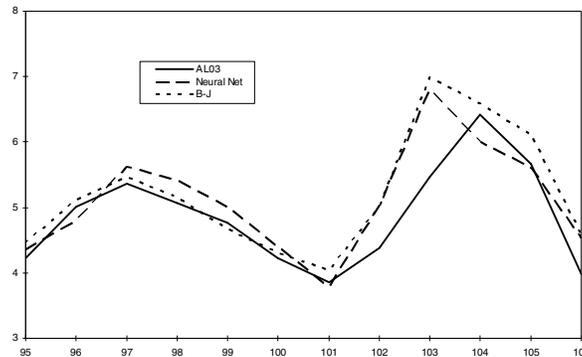
Table 8. Box-Jenkins y Artificial Neural Nets models

## RESULTS FROM THE AL\_SERIES

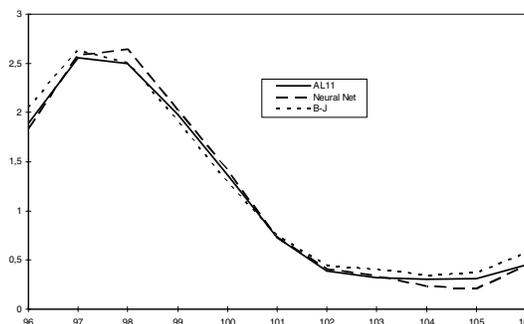
Table 9 shows factory variances for the different methods and the different series. We are this using real data to validate. Such data could not be used with GA in the real world as they would not be available. However, they are presented her to show that even with real data factory variance can be greater than with forecasting methods if Genetic Algorithms are used, as is demonstrated by four of the five series for AL (i.e., all except AL04, which is highlighted in italic, which is also used for similar results such as BJ02, BJ08, UN01, UN06 y UN08). RNA means artificial neural networks methodology; 1-1 Model, Sterman's method (demand maintaining); RAT-MAD-AL simple methods of forecasting as random walk with trend, moving average of 5 periods, or exponential smoothing; of this methods will be used that which better MAPE obtained.

	AL03	AL04	AL09	AL11	AL12
REAL DATA	4038.58	<i>1704.56</i>	5567.78	<b>4846.07</b>	1379.08
RNA	1950.35	1838.18	4262.00	<b>8185.90</b>	923.22
B-J	<i>1799.35</i>	2943.98	<i>2187.33</i>	<b>3888.56</b>	3308.97
1-1 MODEL	8419.11	11081.59	13586.75	<b>1299.02</b>	8108.79
RAT-MAD-AL	2543.55	4357.87	6578.93	<b>6457.97</b>	8323.67

Table 9



Graph 7



Graph 8

Graphs of the forecasts for the different series and real data using both RNA and BJ are described next. The AL03 and AL11 series provide an example. The latter series is the only one of the nineteen where the 1-1 model strategy has proved to provide better

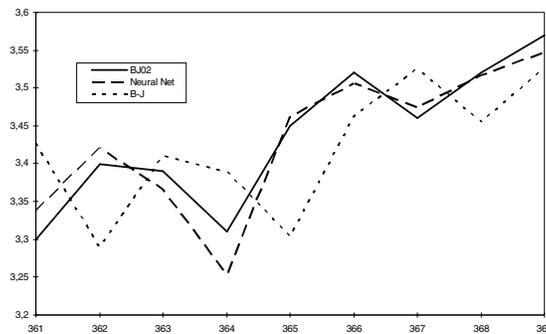
outcomes, as can be seen in table 9. This is because it is a series in which demand falls, which means more stocks. All the levels eliminate this stock and variance with this model is lower than if advanced forecasting methods were applied. Applying simple RAT-MAD-AL forecasting techniques always produced worse results than the 1-1 model, and so they are not considered in later examples, where results are equally inferior.

#### BJ\_ SERIES RESULTS

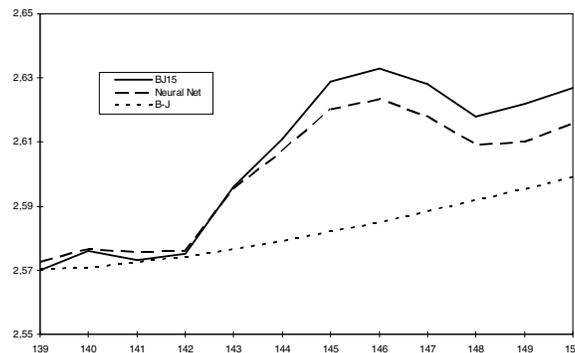
	BJ02	BJ06	BJ08	BJ15
REAL DATA	1666.40	5751.47	2619.64	3531.72
RNA	1956.77	5332.69	5348.18	1033.31
B-J	2560.40	3652.33	5483.45	1571.39
1-1MODEL	20779.69	6587.36	10722.67	4851.90
RAT-MAD-AL	3482.76	7334.76	11256.89	5345.96

Table 10

As table 10 shows, the lowest factory variances were provided by neural networks, with the exception of the BJ06 series, - the Wolfer sunspots. The dynamics are captured far better using networks in the other three series, which were of an economic nature. Graphs 9 and 10 document forecasting results using Box-Jenkins, networks, and the real series.



Graph 9



Graph 10

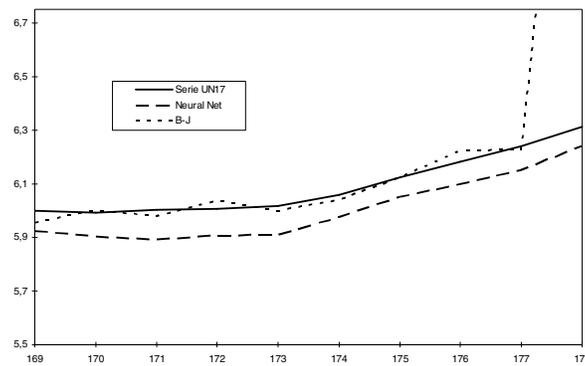
Particular mention should be made of the BJ15 series, as it corresponds to a warehouse's real sales. It can be seen that the best forecasts come from Neural networks, the variance of the manufacturing would be taken by applying the Genetic Algorithm with 1033 units.

RESULTS OF THE UN\_ SERIES

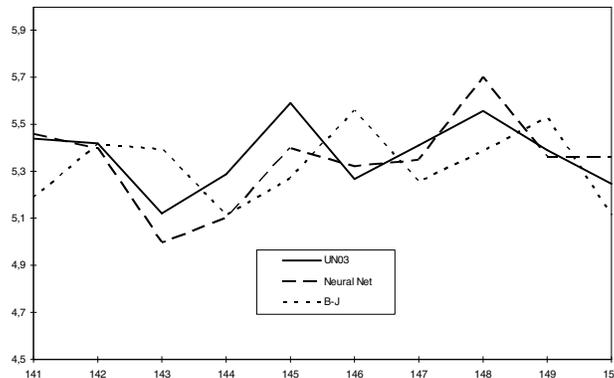
	UN01	UN02	UN03	UN04	UN05
REAL DATA	<b>2097.49</b>	8669.76	5194.00	14257.04	4361.04
RNA	2836.20	2916.61	7356.67	7356.67	18293.76
B-J	2276.89	3316.00	2724.80	2597.16	4022.69
1-1 MODEL	18477.33	10759.96	16675.78	8655.11	11780.40
RAT-MAD-AL	22588.64	12684.84	18956.60	15670.35	12457.98
	UN06	UN07	UN08	UN17	UN21
REAL DATA	<b>876.40</b>	1623.49	<b>2205.64</b>	1632.09	6395.01
RNA	6832.96	709.00	7375.43	1125.64	7359.69
B-J	2636.44	1659.50	5364.36	879.76	5029.36
1-1 MODEL	6850.44	7553.07	17410.71	7165.07	8972.49
RAT-MAD-AL	8568.32	9583.68	18467.34	9468.02	9456.34

Table 11

The results of the series simulated of Reilly (1980) presented in the table 11, it leaves the methodology of Box-Jenkins better than that of nets, this is because these series were simulated thinking basically of this method. The UN17 and UN03 have been included as examples. The results as they relate to the Bullwhip effect are always better with the Genetic Algorithm than with the 1-1 model, because it reduces the stockout in most of the studied cases, except in the series AL11 because it is a decreasing one.



Graph 11



Graph 12

## CONCLUSIONS

This study describes an application of a multi-agent methodology to reduce the Forrester Effect on a supply chain. One part of the agents is a genetic algorithm, which runs on a number of different platforms, and which will take the decision to manufacture, dispatch or take delivery of certain quantities of products according to which forecasting methods were more adequate. The case described has four levels of supply, and the Forrester Effect has been shown to decrease when advanced forecasting tools are applied.

## BIBLIOGRAPHY

- Abraham, B. Ledolter J. *Statistical Methods for Forecasting*. 1983 (Wiley, New York).
- Aviv Y., A. Time-series framework for supply chain inventory management. *Operations Research*, 2003, **51** (2), 210–227.
- Aviv Y., A. Gaining benefits from joint forecasting and replenishment processes. *Manufacturing and Service Operations Management*, 2002, **4** (1), 55–74.
- Aviv Y., A. The effect of collaborative forecasting on supply chain performance. *Management Science*, 2001, **47** (10), 1326–1343.
- Baganha M., M. Cohen M. The stabilizing effect of inventory in supply chains. *Operations Research*, 1998, **46**, 572–583.
- Blanchard, O. J. The production and inventory behavior of the American automobile industry. *Journal of Political Economy*, 1983, **91** 365–400.
- Blinder, A. S. Can the production smoothing model of inventory behavior be saved? *Quarterly Journal of Economics*, 1986, **101** (3) 431–454.
- Box, G. E. P.; Jenkins, G. M. *Time Series Analysis: Forecasting and Control*. 1970, San Francisco: Holden Day.
- Cachon, G. P. Managing supply chain demand variability with scheduled ordering policies. *Management Science*, 1999, **45** (6) 843–856.
- Chen, F., Drezner, Z., Ryan J., Simchi-Levi D. Quantifying the bullwhip effect in a simple supply chain: the impact of forecasting, lead-times, and information. *Management Science*, 2002, **46** (3) 436–443.
- Childerhouse P., Disney S. M., Towill D. R. Tailored toolkit to enable seamless supply chains. *International Journal of Production Research*, 2004, **42** (17) 3627–3646.
- Dejonckheere, J., Disney, S. M., Lambrecht M. R., D.R. Towill D. R. Measuring and avoiding the bullwhip effect: a control theoretic approach. *European Journal of Operational Research*, 2003, **147** (3) 567–590.
- Dejonckheere, J., Disney S. M., Lambrecht M. R., Towill D. R. The impact of information enrichment on Bullwhip effect in supply chain: A control engineering perspective. *European Journal of Operational Research*, 2004, **153**, 727–750.
- Disney S. M., The production and inventory control problem in vendor manager inventory supply chain. Ph. D. Thesis, 2001, Cardiff Business School, Cardiff University, UK.
- Disney S. M., Towill D. R. Transfer function analysis of forecasting induced bullwhip in supply chain. *International Journal of Production Economics*, 2002, **78**, 133–144.
- Disney S. M., Towill D. R. The effect of Vendor Managed Inventory (VMI) dynamics on the Bullwhip effect in supply chain. *International Journal of Production Economics*, 2003, **85**, 199–215.
- Disney S. M., Towill D. R. On the Bullwhip and inventory variance produced by an ordering policy. *Omega*, 2003, **31**, 157–167.
- Disney S. M., Grubbstrom R. W. Economic consequences of a production and inventory control policy. *International Journal of Production Research*, 2004, **42** (17) 3419–3431.

- 1  
2  
3 Disney S. M., Naim M. M. Potter A. Assessing the impact of e-business on supply chain  
4 dynamics. *International Journal Economics*, 2004, **89**, 109-118.  
5 Disney S. M., Towill D. R., Velde W. Variance amplification and the golden ratio in  
6 production and inventory control. *International Journal of Production Economics*, 2004,  
7 **90**, 295-309.  
8 Drezner Z, Ryan J, Simchi-Levi D. *Quantifying the bullwhip effect: the impact of*  
9 *forecasting, lead time and information*. Working Paper, Northwestern University,  
10 Evanston, IL, 1996.  
11 Forrester, J. W. *Industrial dynamics*, MIT Press, 1961, Cambridge, MA.  
12 Cachon G. P. Lariviere, M. A. Capacity choice and allocation: strategic behavior and  
13 supply chain performance. *Management Science*, 1999, **45** (8), 1091-1108.  
14 Graves, S. C. A single-item inventory model for a nonstationary demand process.  
15 *Manufacturing & Service Operations Management*, 1999, **1** 50-61.  
16 J.D. Serman, J. D. Modeling managerial behavior: misperceptions of feedback in a  
17 dynamic decision making experiment. *Management Science*, 1989, **35** (3) 321-339.  
18 Kahn, J. Inventories and the volatility of production. *The American Economic Review*,  
19 1987, **77**, 667-679.  
20 Kaminsky P, Simchi-Levi D. *A new computerized beer game: a tool for teaching the*  
21 *value of integrated supply chain management*, POMS series in technology and  
22 operation management, Global supply chain and technology management, USA, 1998,  
23 1.  
24 Kelle P. Milne, A. The effect of (s,S) ordering policy on the supply chain. *International*  
25 *Journal of Production Economics*, 1999, **59**, 113-122.  
26 Makridakis, S. (1993): Accuracy Measures: Theoretical and Practical Concerns.  
27 *International Journal of Forecasting*, 9 527-529.  
28 Reilly D. P. *Experience with an automatic Box-Jenkins Modelling in Time Series*  
29 *Analysis*, O. D. Anderson, Amsterdam: North-Holland, 1980.  
30 Slutsky, E. *The summation of random causes as the source of cyclic processes*.  
31 *Econometrica*, 1937, **5**, 105-46.  
32 Serman, J. Modelling managerial behaviour: Misperceptions of feedback in a dynamic  
33 decision making experiment, *Management Science*, 1989, **35**(3), 321-339.  
34 Serman, J. D. Teaching takes off, flight simulators for management education. *OR/MS*  
35 *Today*, 1992, **19** (5) 40-44.  
36 Serman, J. D. *The beer distribution game*. In: J. Heineke and L. Meile, Editors, *Games*  
37 *and exercises for operations management*, Prentice Hall, Englewood Cliffs, NJ, 1995,  
38 101-112.  
39 West, K. D. A variance bounds test of the linear quadratic inventory model. *Journal of*  
40 *Political Economy*, 1986, **94** (4) 374-401.  
41 Wold, H. *A study in the analysis of stationary time series*. Estocolmo: Almgrist &  
42 Wiksell 1938.  
43 Yule, G.U. Why do we sometimes get nonsense-correlations between time series? A  
44 study in sampling and the nature of time series. *Journal of Royal Statistical Society*,  
45 1926, **89**, 1-64.  
46  
47  
48  
49  
50

## 51 **APPENDIX: Explanation of pseudocode**

52  
53 In order to clarify the computation of the fitness, there are two kinds of Supply Chain  
54 levels: the Shop retailer who faces the demand and which order to the retailer is  
55 passed along the other levels, which takes the shop retailer information (detail) as a  
56 input in the GA evolution process.  
57  
58  
59  
60

SHOP RETAILER

```

sub fitdetall {
  w = .5+rand(0:1); # random between 0.5 and 1.5
  media=0;
  for z (p-1..período actual p){ # loop from z=p-1 up to z=p
    x=demanda[z]-invfdetall[z]+rstockdetall[z];
    # x = demand - final inventory retail shop+ stockout retail shop (all of the
    period z)
    y = abs(w*x); # y = absolute value of random per x
    media += y; # acumulated value of y over z periods
  }
  detall=media/2; # final value, from 2 periods, which could be ordered to retailer
  # (this data is exported)
  return 1/detall; # the minor, the better, thus the inverse is taken

```

RETAILER

```

sub fitmino {
  w = .5+rand(0:1);
  media=0;
  for z (p-1..p){
    x=pidedetall[z]-invfmino[z]+rstockmino[z];
    # x = order from - final inventory + stockout (all from retailer, period z)
    y = abs(w*(x+detall)/2);
    # y = random between 0.5 and 1.5 per the mean between x and the what the
    shop retailer has orderered in the actual period (variable imported)
    media += y; # accumulated of y, 2 periods
  }
  mino=media/2; # mean of y, 2 periods
  return 1/mino; # return the inverse, the maximum is the fittest
}

```



UNIVERSIDAD DE OVIEDO

**Dr. David de la Fuente**  
Escuela Politécnica Superior de Ingeniería  
Campus de Viesques, E-33204 Gijón (SPAIN)

Phone: +(34)985.18.21.47  
e-mail: david@uniovi.es

Fax: +(34)985.18.20.10  
<http://gio.uniovi.es>

Gijón, 13 October 2005

Dear Carmela Valentine,

Please find attached the second version of our paper “The application of distributed intelligence to reduce the bullwhip effect”, and a letter explaining the changes made. We think we have considered the referee’s comments that you sent to us in your letter.

In relation to the comments about the papers we have considered all their indications, we have created an explanatory appendix of the fitness functions like the referee suggested to us. We have also changed the errors of the pages 5, 8 and we made changes in the pages 15, 16. As for the effect of the hardware and the used software, it is possible that both influence in the times of execution.

I look forward to hearing from you,

Fdo. David de la Fuente