Département de gestion
des opérations et de la logistique

Teaching inventory management using IMSim V.0.0.1
Teaching note: *www.nineties-memorabilia.com* scenario

Jorge E. Mendoza
jorge.mendoza@hec.ca

Introduction:

Inventory management (IM) is a key concept in logistics. Operations management and logistics courses, both at undergrad and post-graduate levels, typically include (at least) a chapter introducing students to this topic. Inventory management is typically taught using a combination of lectures, in-class exercises and (in rare cases) spread-sheet simulation. This teaching approach has proven effective. However, students often feel that they do not have the chance to apply what they learn in a concrete practical situation. This pedagogical activity aims to help filling that void.

The activity presents students with the case of *www.nineties-memorabilia.com* a (fictional) online retailer specialized on 90's memorabilia. The company buys products from different retailers and resells them to their customers through their internet site. The scenario studied in this activity focuses on a particular product family: memorabilia from the 90's most popular sit com, Seinfeld. The scenario includes 7 products with different characteristics (e.g., type of demand, lead time, and price structure). At the beginning of the activity, students are given an excel spread sheet (`nineties-memorabilia.com.xlsx`) holding the daily sales for the 7 products in the last three years. The students must analyze the data, select appropriate inventory management policies for each product, and compute the optimal parameters for those policies. To evaluate their policies, the students must rely on IMSim V.0.0.1, as stochastic simulation software developed at HEC Montréal. In a nutshell, IMSim V.0.0.1 allows students to define IM policies for a (predefined) set of products and simulate the operation the policies during a given time horizon (e.g., one year). The application then reports classical IM metrics such as: service level, fill rate, number of placed orders, average inventory levels, total ordering cost, total holding cost, and total acquisition cost. It also provides students with a detailed view of inventory levels and IM decisions for each simulated day.

This pedagogical activity has three main learning outcomes. After the activity the students are expected to be able to:

- Select the best inventory management policy for a product based on its characteristics and their (previous) knowledge of IM models.
- Estimate the optimal parameters for different types of IM policies based on the product's characteristics.
- Compare alternative policies for the same product based on the performance metrics delivered by the simulation software.

The `www.nineties-memorabilia.com` scenario

`www.nineties-memorabilia.com` is a web retailer specialized in 90's memorabilia. The company's portfolio of products includes 90's popular toys (e.g. G.I Joes, Barbies, Lego sets), CDs, VHS cassettes, videogames and consoles, and all kinds of TV show and movie mementos. One the company's most popular product line is memorabilia from the popular sit com Seinfeld. In particular, the company sells 7 items that are best sellers among the show's diehard fans. These items are bought from 5 different suppliers. The following tables present the description of the products. The last row in the tables (in grey) gives information that is blind to the students.

Coffee table book (about coffee tables)

Supplier: Pendant Publishing
Ordering cost: $20.0
Holding cost rate: 10.0%
Target service level: 100.0%
Lead time: 0 days
Initial stock: 0 units
Price: $30.00

Demand : constant-deterministic with a rate of 195 units/day

Himalayan Walking Shoes

Supplier: J. Peterman
Ordering cost: $15.0
Holding cost rate: 12.0%
Target serv. level: 100.0%
Lead time: 0 days
Initial stock: 0 units
Price: $120

Demand : constant-deterministic with a rate of 3 units/day

King Edward VIII's wedding cake replica

Supplier: J. Peterman
Ordering cost: $15.0
Holding cost rate: 12.0%
Target serv. level: 100.0%
Lead time: 0 days
Initial stock: 0 units
Price: $95

Demand : constant-deterministic with a rate of 4 units/day

Soup Nazi bubble head

Supplier: Kramerica Industries
Ordering cost: $20.0
Holding cost rate: 12.0%
Target serv. level: 100.0%
Lead time: 0 days
Initial stock: 0 units
Price: 100

Demand : constant-deterministic with a rate of 12 units/day

Puffy Shirt replica	
	**Supplier:** Vandelay Industries **Ordering cost:** $150.0 **Holding cost rate:** 10.0% **Target serv. level:** 90.0% **Lead time:** 2 days **Initial stock:** 800 units **Price:** $100
Demand : normal distribution with a mean of 400 units/day and a std. deviation of 50 units	

SP2000 Slicer miniature	
	**Supplier:** Kruger Industrial Smoothing **Ordering cost:** $20.0 **Holding cost rate:** 20.0% **Target serv. level:** 100.0% **Lead time:** 0 days **Initial stock:** 0 units **Price:** $5 (between 1 and 649 units), 4.5 (between 650 and 999 units), 3.9 (1000 units or more)
Demand : constant-deterministic with a rate of 28 units/day	

Urban Sombrero	
	**Supplier:** J. Peterman **Ordering cost:** $15.0 **Holding cost rate:** 12.0% **Target serv. level:** 100.0% **Lead time:** 0 days **Initial stock:** 0 units **Price:** $225
Demand : constant-deterministic with a rate of 2 units/day	

Students have access to 3 years of sales data for each product. The data is available in a spread sheet named `nineties-memorabilia.xsls`.

Understanding and using IMSim V.0.0.1

IMSim V.0.0.1 is a stochastic simulation software developed at HEC Montréal specifically for this pedagogical activity. The application is available in French, English, and Spanish. The instructor can adjust the language according to requirements of the class simply by modifying a parameter on the configuration file (`params.xml`) that can be found in the `config` folder. Figure 1 shows the `params.xml` file setting the language to English. To change the language to French or Spanish, the user only needs to change the value of parameter language from `en` to `fr` or `es`.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
<entry key="sim_output_file">./simulations/simlog_simulation</entry>
<entry key="sim_output_override">true</entry>
<entry key="language">en</entry>
<entry key="seed">9</entry>
<entry key="randomize">false</entry>
<entry key="scenario">nineties-memorabilia</entry>
</properties>
```

Figure 1: `params.xml` configuration file

IMSim V.0.0.1 is supplied with a single scenario (the one discussed in this teaching note). The file holding the scenario can be found in the folder named `scenarios`. Figure 2 shows an extract the file holding the data for the scenario discussed in this note (i.e., `nineties-memorabilia.xml`). Instructors are free to create new scenarios for their classes. The files holding scenario data **must** be placed in the `scenarios` folder and they **must** comply with a predefined format. A complete guide on how to write scenario files will soon be made available to instructors. To change the scenario, the instructor only needs to change the value of parameter `scenario` in the `params.xml` file. The value of the parameter **must** correspond to the name of the scenario file (without the `.xml` extension). Figure 1 shows the configuration file defining `nineties-memorabilia.xml` as the file holding the parameters for the scenario.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<scenario>
    <periods>365</periods>
    <products>7</products>
    <suppliers>5</suppliers>
    <supplier_list>
        <supplier>
            <name>Kramerica Industries</name>
            <fees>30</fees>
        </supplier>
        <supplier>
        <supplier>
        <supplier>
        <supplier>
    </supplier_list>
    <product_list>
        <product>
            <name>Coffee table book</name>
            <supplier>Pendant Publishing</supplier>
            <demand type="CONSTANT">
                <rate>195</rate>
            </demand>
            <prices>
                <interval>
                    <lb>1</lb>
                    <ub>inf.</ub>
                    <price>30</price>
                </interval>
            </prices>
            <co>20</co>
            <ch>0.1</ch>
            <sl>1</sl>
            <lt>0</lt>
            <init_stock>0</init_stock>
        </product>
        <product>
```

Figure 2: partial view of the `nineties-memorabilia.xml` file holding the data for the scenario discussed in this note

Using the application

The graphical user interface presents the user with 4 tabs: `Product`, `Suppliers`, `Policies`, and `Simulation`. As the name suggest, in the `Product` tab students can access product information. Figure 2 shows an example. The `Suppliers` tab presents the list of suppliers and their parameters. Table 1 describes the parameters for products and suppliers.

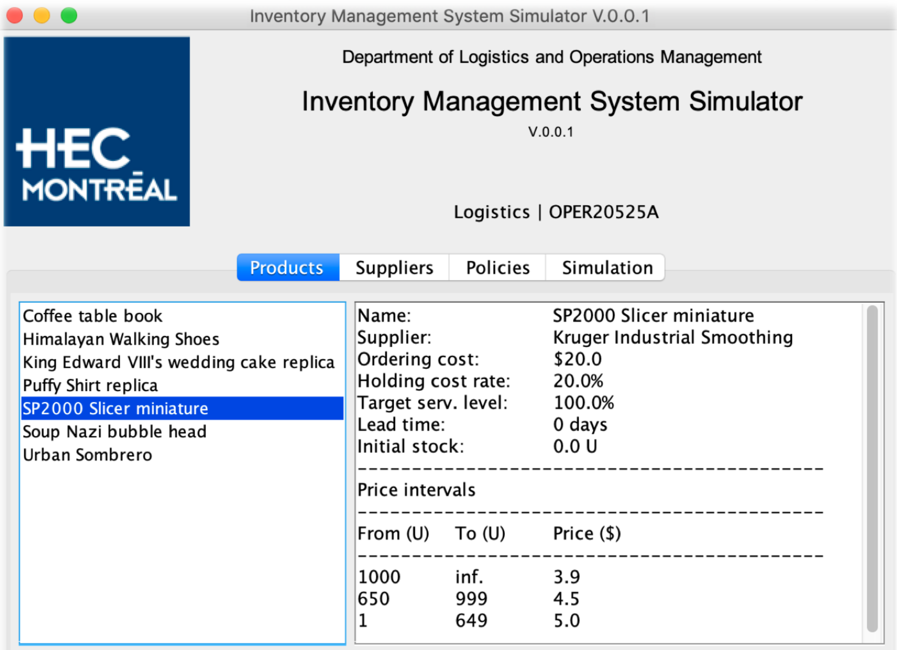

Figure 3: IMSim V.0.0.1 graphical user interface

Products	
Name	Name of the product.
Supplier	Name of the supplier of the product.
Ordering cost ($)	Cost of ordering the product. The company incurs in this cost every time they order the product (independently of the ordered quantity).
Holding cost rate (%)	The cost of holding to one unit of the product for one year as a proportion of the price of the product. For instance if the cost of a product is $100 and the holding cost rate is 12%, keeping a unit of that product in inventory for a year would cost the company $12. In the simulation, the holding costs are computed on the average inventory held for each product each day. More specifically, the holding cost for product *p* on day *t*, denoted as $C_{h_{p,t}}$, is computed as follows: $$C_{h_{p,t}} = \left(\frac{holding\ cost\ rate_p \times price_p}{T}\right) \times \bar{s}_{p,t}$$ where $price_p$ is the price of product *p*, *T* is the number of simulated days (365 in the `www.nineteens-`

	`memorabilia.com` scenario), and $\bar{s}_{p,t}$ is the average stock for product *p* during day *t*. The total holding cost for product *p* is then equal to $\sum_{t=1}^{t=T} C_{h_{p,t}}$.
Target fill rate	Target fill rate for the product. For instance, if the target fulfillment rate for a product is 95% and the total demand for the product during the simulation is 1000 units, the company expects to have an inventory management policy that allows them to serve at least 950 units of that demand. In the simulation, the lost sales for product *p* on day *t* are computed as follows: $$lost\ sales_{p,t} = -1 \times Min\{s^I_{p,t} + s^R_{p,t} - d_{p,t}, 0\}$$ were $s^I_{p,t}$ is the stock for product *p* at the beginning of day *t*, $s^R_{p,t}$ is the stock for product *p* received (from previously placed orders) during day *t*, and $d_{p,t}$ is the demand for product *p* in day *t*. At the end of the simulation the fulfillment rate for product *p* is computed as: $$service\ level_p = 1 - \frac{\sum_{t=1}^{t=T} lost\ sales_{p,t}}{\sum_{t=1}^{t=T} d_{p,t}}$$
Target Serv. Level	Target cycle service level for the product. The fraction of replenishment cycles that end with all the customer demand being met. In IMSim V.0.0.1, a replenishment cycle is the number of periods between two consecutive orders.
Lead time (days)	The time needed to replenish the product. In the simulation, orders are assumed to be placed at the **end** of the day. However, if the lead time for a product is 0, an order placed at the end of day *t* is assumed to be received during *t* and the new stock can be used to cover the demand of *t*. If the lead time is greater than 0, the stock corresponding to an order placed in day *t* is assumed to become available during day *t* + *lead time*. For instance if the lead time of a product is 15 and an order is placed in day 2, the fresh stock becomes available at some point during day 17, but it can be used to cover the entire demand of that day.
Initial stock (units)	The stock available for product *p* at the beginning of the simulation.
Price intervals	The price structure for the product. IMSim V.0.0.1 allows products with both a fix price and a price that varies according to the ordered quantities (i.e., products with quantity discounts). The latter is modeled using price intervals. A price interval is defined by a triplet (`From, To, Price`), where `From` is the number of units marking the beginning of the interval, `To` is the number of units marking the end of the interval, and `Price` is the price payed for each unit of an order, if the order

	quantity lays between `From` and `To`. For instance, if the price intervals for a given product are defined as {(1, 999, $150), (1000, inf., $100)} the cost for a unit of the product is $150 if the ordered quantity lays between 1 and 999 units, and $100 if the latter is larger than or equal to 1000.

Suppliers	
Name	The name of the supplier.
Fees ($)	The administrative fees charged by the supplier for processing an order. These fees are added to the ordering costs on top of the individual ordering cost for each product included in an order. For example, if we order product *p* from supplier *s*, the total ordering cost for a given order would be: the ordering cost for *p* + the administrative fees of *s*. On the other hand, if an order contains products *p* and *q*, the total ordering cost for a given order would be: the ordering cost for *p* + the ordering cost for *q* + the administrative fees of *s*. During the simulation, the administrative fees for a given order are equally distributed among the products included in the order independently of their ordered quantities. For instance, if the ordering cost for product *p* is $10, the ordering cost for product *q* is $20, the administrative fees of supplier *s* are $40, and we jointly order *p* and *q* from *s*, the total ordering cost for *p* will be $30 (=$10+$40/2) and that of *q* will be $40 (=$20+$40/2).

Table 1: description of the product and supplier parameters in IMSim V.0.0.1

The `Policies` tap allows the user to define IM policies for one or more products (i.e., joint replenishment policies). IMSim V.0.0.1 handles two types of policies: continuous review policies (CRPs) with a fixed reorder point and a fixed order quantity, and periodic review policies (PRPs) with a fixed review interval and a variable order quantity. For theoretical background on these policies, the instructor is invited to read the pedagogical note in [1].

> **Teaching tip**: provide your students with a copy of [1] and ask them to read the document before the session. All the IM models used in this activity are explained in that pedagogical note.

Defining IM policies

After selecting the type of policy, click on the “`Add new inventory control policy`” button. Depending on the type of policy a different type of pop-up menu will emerge. Figure 4 shows the menu for CRPs. Using the combo box, select the product for which you want to define the policy. Note that only products for which no policy has been defined will show on the menu. Next, set the two parameters for the CRP: the reorder point and the order quantity. The example shown in Figure 4, defines a reorder point of 100 units and an order quantity of 500 for the SP2000 Slicer miniature. During the simulation, every time the stock reaches 100 units, a new order for 500 units is placed. To save the CRP, click on the “`Save policy and close`” button. To abort the operation simply click on `Cancel`. It is worth noting that in IMSim V.0.0.1 CRPs are limited to a single product.

Teaching tip: Defining joint CRPs for multiple products is impractical, because each product is likely to reach the reorder point at a different point in time. While presenting this menu, discuss with your students this issue and let them brainstorm on potential ways to overcome it.

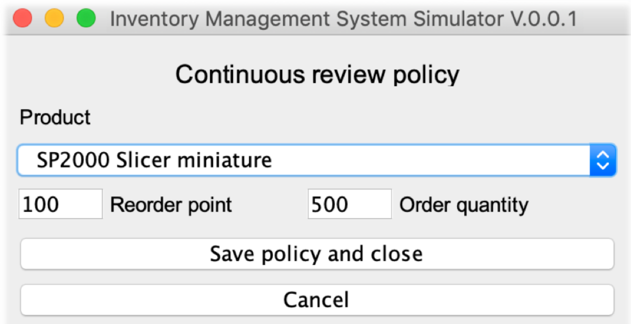

Figure 4: Continuous review policy menu

The newly defined policy will be displayed on the bottom panel of the `Policies` tap using the following convention:

```
Continuous review | PRODUCT_NAME (REORDER_POINT | ORDER_QUANTITY)
```

To delete a policy, select the policy on the bottom panel in the `Policies` tap and click on the "`Delete control policy`" button. Please note that IMSim V.0.0.1 does not allow users to edit (i.e., change the parameters of) an existing policy.

Figure 5 shows the pop-up menu for defining PRPs. Contrary to CRPs, PRPs can be jointly defined for multiple products (as long as they are bought from the same supplier). To add a product to a PRP, select it from the combo box, set the value for its target stock, and click on the "`Add to policy`" button. To add more products to the policy, repeat the procedure. After the first product is added to the policy, the combo box will only display the products that are sold by the same supplier. After adding all the products to the policy, set the cycle length (also known as review interval) field and click on the "`Save policy and close button`". According to the example displayed in Figure 5, during the simulation, the application will check the stocks for the Himalayan Walking Shoes and the Urban Sombrero every 20 (simulated) days and orders enough of each product to reach their corresponding target stocks (700 and 300 units, respectively).

The newly added PRP will be displayed on the bottom panel of the `Policies` tab using the following convention:

```
Periodic review (INTEVAL) | PRODUCT_1_NAME (TARGET_STOCK),
PRODUCT_2_NAME (TARGET_STOCK), ..., PRODUCT_P_NAME (TARGET_STOCK)
```

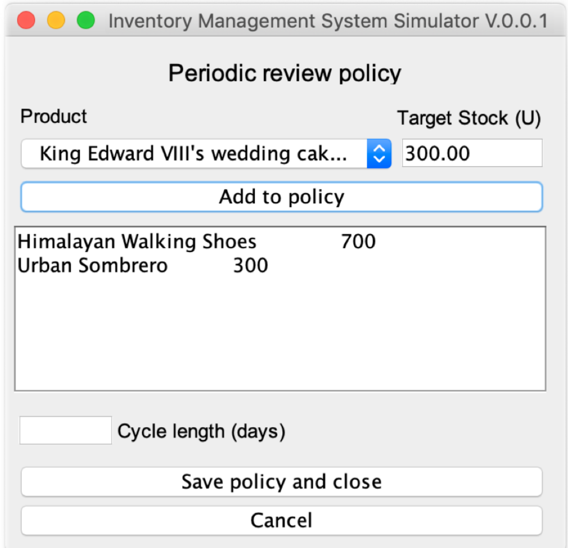

Figure 5: Periodic review policy menu

Figure 6 shows the Policies tab after adding the CRP and the PRP discussed in this section.

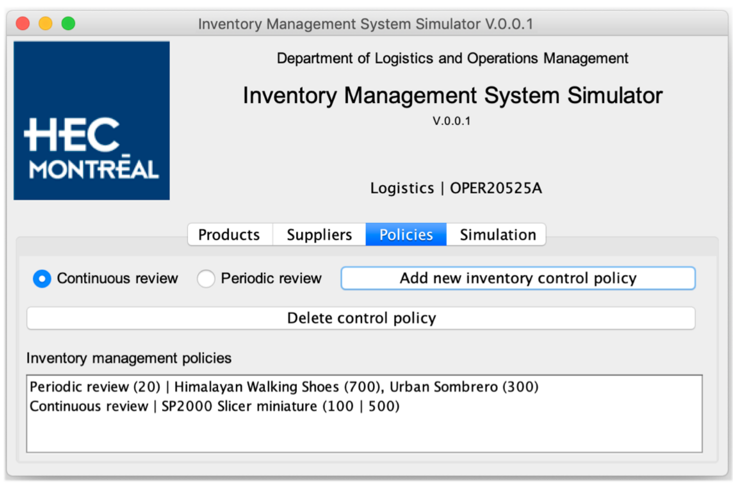

Figure 6: The `Policies` tab after adding a CRP and a PRP

The Simulation tab

The `Simulation` tab allows the user to evaluate the defined policies using discrete event simulation. To simulate the operation of a policy, select the policy from the top list and click on the "`Evaluate policy button`". The application will run the simulation and report, for each product included in the policy, i) the performance metrics, and ii) the detailed simulation data. The performance metrics include: the product fill rate, the cycle service level, the total number of orders, the average daily stock, the total ordering cost (C_o), the total holding cost (C_h), the total acquisition cost (C_a), and the total cost (CT). Figure 7 shows the evaluation of the CRP defined in the previous section.

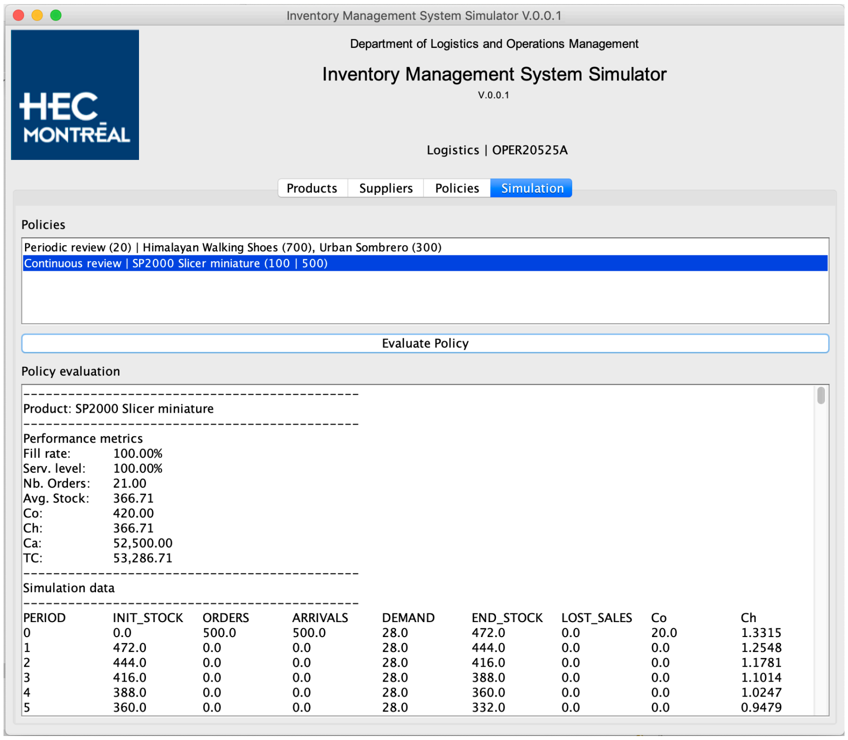

Figure 7: evaluation of a CRP

The detailed simulation data shows for every (simulated) day the following fields: stock at the beginning of the day (`INIT_STOCK`), the number units ordered during the day (`ORDERS`), the number of received during the day (`ARRIVALS`), the demand for the day (`DEMAND`), the stock at the end of the day (`END_STOCK`), the unfilled demand (`LOST_SALES`), the ordering cost associated to orders placed during the day (`Co`), the cost of holding inventory for that day (`Ch`). To help understanding this data, let us examine the case for day 0 in the example displayed in Figure 7. The initial stock for the product is 0. Since the policy fixes the reorder point to 100, an order for 500 units is immediately placed. Because the lead time of the product is 0, the order is received right away and the fresh inventory can be used to serve the 28 units of demand observed on the day. The final stock is then 472 (=500-28). Since the ordering cost for the product is $20 and the supplier applies no administrative fees, the total ordering cost for the day is $20. The holding cost is slightly more difficult to understand. Since the order is placed and received at the beginning of the day and the demand is 28 units, the average inventory is 486 (=472+28/2). The holding cost rate for the product is 20% and its price is $5.0 -- note that the order quantity belongs to the (0, 649, $5.0) price interval. Therefore, holding a unit of inventory for a day costs $0.0027 and the total holding cost of the day is 1.3315 (=0.0027 x 486).

Teaching tip: the *IMSim V.0.0.1 short user guide for students* presents an extract of this document including this specific section. Give a copy of the guide to your students and ask them to read it before the session. Demonstrate the use of the software on the board before starting the activity. An easy demo is to repeat the examples discussed on this section.

The activity

This section provides suggestions on how to run this pedagogical activity in class. The activity is designed to take between 1.5 and 2 hours depending on how many products the instructor wants the students to analyze and how much time s/he wants to give them to explore possible solutions by themselves.

Before the session

IMSim V.0.0.1 was developed in Java; it is therefore multi-platform (e.g., Mac OS, Windows, Linux). The only requirement to run the application is to have the Java Runtime Environment (JRE) – version 8 or superior – installed in your system. Most computers already have a version of the JRE installed. It is, however, important that you make sure that your computer is equipped with the right version of the JRE before the session.

To check if Java is installed in your computer, follow these instructions

1. Start the Command Prompt (Windows) or Terminal (Mac). If you do not know how to accomplish this, you can find quick tutorials for Windows and Mac OS by clicking on the corresponding links[1].
2. Type the following command and press enter: `java -version`.

If Java is correctly installed in your computer, you should obtain an output similar to that displayed in Figure 1.

```
Jorges-MBP:~ jemendoza$ java -version
java version "10.0.2" 2018-07-17
Java(TM) SE Runtime Environment 18.3 (build 10.0.2+13)
Java HotSpot(TM) 64-Bit Server VM 18.3 (build 10.0.2+13, mixed mode)
Jorges-MBP:~ jemendoza$
```

Figure 1: executing the `java -version` command

If you your computer does not recognize the `java -version` command, Java is not installed on your system. You can download the current version of the JRE for free from this address: https://www.java.com/download/. To install the JRE in your computer simply follow the instructions. After completing the installation, run the test again to make sure that your computer is ready for the activity.

> **Teaching tip**: ask your students to download and install the current version of the JRE before the session. This will save time and troubles during class-time.

Once you have installed and tested the JRE, you can download IMSim V.0.0.1 from the following link: https://github.com/jemendoza/IMSim/releases/download/V.0.0.1/IMSim-V.0.0.1.zip. Unzip the contents of the .zip file to an easy-to-remember location in your file system. To launch the application, simply double click on the `IMSim-V.0.0.1.jar` file.

> **Teaching tip**: ask your students to download the application and make sure it launches correctly before the session. This will save time and troubles during class-time.

[1] Last access: August 20, 2019

During the session

This section outlines a potential agenda for the class session. This agenda mimics the one followed by the author in the class taught on the senior year of the undergraduate program in business administration at HEC Montréal.

Task	Time	Description	Duration
Distribute data files	Before session	Distribute the nineties-`memorabilia.xlsx and nineties-memorabilia.docx` files to the students through the teaching platform of your institution	-
Read guidelines	Before session	Ask students to read the `nineties-memorabilia.docx` file	15 min.
Meet in groups	During session	Ask students to meet in groups of two. To save some time, you can ask them to form the groups before the session	5 min.
Re-read guidelines	During session	Ask the students to quickly re-read and discuss (with their partners) the `nineties-memorabilia.docx` document.	10 min.
Demand analysis	During session	Ask the students to open the `nineties-memorabilia.xslx` file and analyze the demand for the 7 products. They should identify the demand patterns presented in the product sheets earlier in this teaching note. You can alternatively ask them to complete this task at home before the session. The latter saves time and creates incentive for students to prepare for the activity.	20 min.
Guided exercise 1	During session	Using product "Coffee table book" as an example, walk the students through the process of analyzing demand data, selecting an IM policy, computing the optimal parameters for the policy, and assessing the policy using IMSim.	30 min.
Exercises 2-6	During session	Ask students to repeat the process with the remaining 6 products. Go around the class providing advice to students and engaging them in discussions about their choices.	45 min.
Solution 2-6	During session	Do a product-by-product analysis on the board with the help of the students. For each product, you can select one team and ask them to propose their solution. Discuss the solution with the rest of the class before giving them feedback and showing your own computations.	60 min.

The reminder of this section i) presents support material for the Guided Exercise 1, that is, a detailed analysis for product "Coffee table book" and ii) proposes solutions for the remining 6 products. All the computations discussed in this section are accessible to the Instructor in the `nineties-memorabilia_analysis.xslx` file accompanying this note.

<u>Policy 1: Coffee table book</u>

The Coffee table book has a constant demand with a daily rate of 195 units. This pattern is easy to identify using an MS Excel pivot chart. Starting from sheet `Demands` on the `nineties-memorabilia.xslx` file, use column `PRODUCT` as filter, column `DAY` as the sole axis category, and the sum of the values in column `DEMAND` as the values. Figure 8 shows the resulting chart. The total yearly demand for the product is 71175.

Since the product has a perfectly deterministic and constant demand, we can use the Economic Order Quantity (EOQ) model to estimate the optimal order quantity. The computation is shown below:

$$Q = EOQ = \sqrt{\frac{2 \times C_o \times TD}{C_u \ \times C_h}} = \sqrt{\frac{2 \times 20 \times 71175}{30 \ \times 0.10}} = 974.17 \approx 975$$

where C_o is the ordering cost, TD is the total yearly demand, C_u is the acquisition cost (for a unit), and C_h the annual holding cost rate. Note that the supplier of this product (Pendant Publishing) does not apply administrative fees. Therefore the ordering cost is simply $20 as stated in the product sheet. Note also that this product has a fixed price that is independent of the number of ordered units. The unit acquisition cost is then $30.

Ordering this quantity leads to the following (annual) total cost:

$$TC = \frac{TD}{Q} \times C_o + \frac{Q}{2} \times (C_u \times C_h) + C_u \times TD = \frac{71175}{975} \times 20 + \frac{975}{2} \times 3 + 2\ 135\ 250 = 2\ 138\ 172.5$$

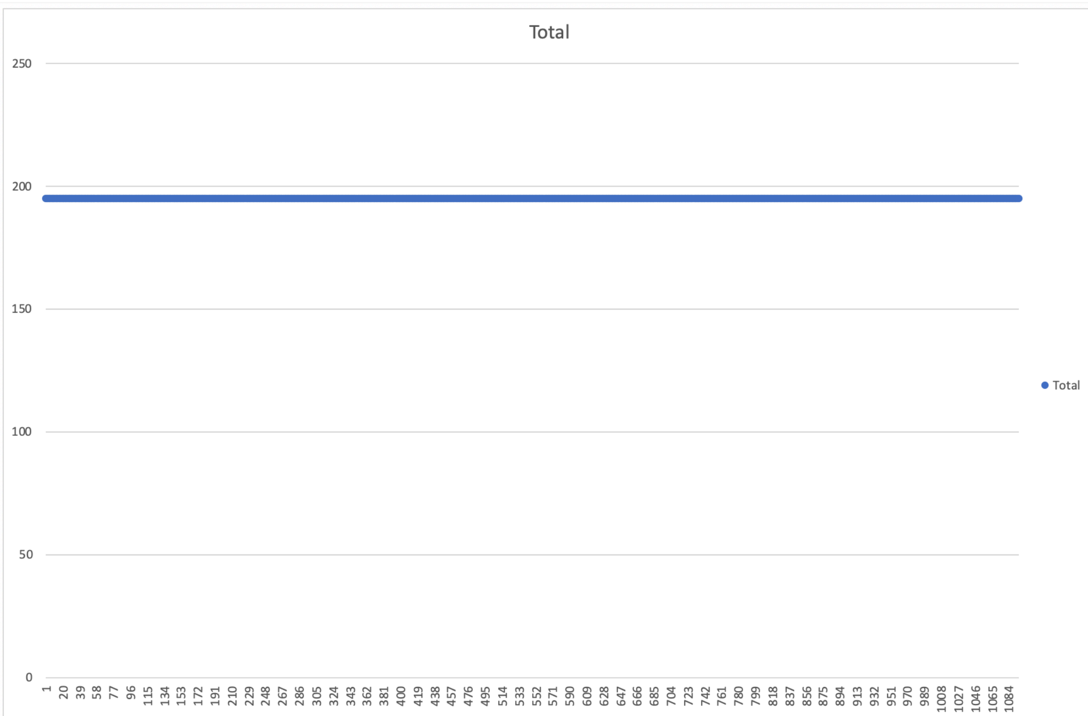

Figure 8: demand analysis for product “Coffee table book”

Now that we have computed the optimal order quantity, we can build an inventory management policy. Since the demand is deterministic and constant, we can easily establish that an order (i.e., 975 units) can cover the demand of exactly 5 days (=977/195). Since the lead time for this product is 0, we can safely place orders every 5 days without keeping safety stock. We can therefore establish a PRP for this product where the target stock is 975 and the revision interval is 5. The video `example1.mov` accompanying this note shows how to define and evaluate such a policy using IMSim V.0.0.1. Figure 9, shows the result of the policy evaluation.

```
---------------------------------------------
Product: Coffee table book
---------------------------------------------
Performance metrics
Fill rate:      100.00%
Serv. level:    100.00%
Nb. Orders:     73.00
Avg. Stock:     487.50
Co:             1,460.00
Ch:             1,462.50
Ca:             2,135,250.00
TC:             2,138,172.50
---------------------------------------------
Simulation data
---------------------------------------------
PERIOD    INIT_STOCK  ORDERS    ARRIVALS  DEMAND    END_STOCK  LOST_SALES  Co      Ch
0         0.0         975.0     975.0     195.0     780.0      0.0         20.0    7.2123
1         780.0       0.0       0.0       195.0     585.0      0.0         0.0     5.6096
2         585.0       0.0       0.0       195.0     390.0      0.0         0.0     4.0068
3         390.0       0.0       0.0       195.0     195.0      0.0         0.0     2.4041
4         195.0       0.0       0.0       195.0     0.0        0.0         0.0     0.8014
5         0.0         975.0     975.0     195.0     780.0      0.0         20.0    7.2123
6         780.0       0.0       0.0       195.0     585.0      0.0         0.0     5.6096
7         585.0       0.0       0.0       195.0     390.0      0.0         0.0     4.0068
```

Figure 9: evaluation of a PRP policy for product “Coffee table book”

Note that the target service level (i.e., 100%) was achieved, and that the simulation of the policy returned the exact ordering, holding, acquisition, and total cost estimated using the theoretical model. Please also notice that this may not be the case for other products because the theoretical model works on a continuous time line, but the simulation works on a discrete time line with a step of one day.

Now that we have found and evaluated the optimal inventory management policy for this product, we can engage the students into a discussion on how sub-optimal policies impact the different performance metrics. We can show students what happens when we order less than EOQ or more than EOQ; what happens when we order more, or less, often than the optimal revision interval of 5 days. Figure 10 shows results for a few sub optimal policies. For reproducibility purposes, the computations are available to the Instructor in file `nineties-memorabilia_analysis.xslx`.

The second column shows a policy where the order quantity (i.e., 1200) is greater than EOQ and the revision interval (i.e., 6 days[2]) is greater than the optimal interval of 5 days. In that case the number of orders reduces from 73 to 61, resulting on a lower ordering cost ($1220 vs. $1460). On the other hand, since the order quantities are higher and the revision period is not exactly Q/195 (remember that the simulator works on a discrete timeline with a step of 1 day), more units are held in stock during the simulation. Consequently, the holding cost increases from $1462.50 to $2397.76. For the same reason, the total number of units acquired during the simulation is larger. As a result, the total cost increases 0.62% with respect to the optimal policy.

[2] The revision interval was computed as $\lfloor 1200/195 \rfloor$

INVENTORY POLICIES

			Q=EOQ | P=P*	Q > EOQ | P>P*	Q<EOQ | P<P*	Q=EOQ | P>P*
Parameters		Mean (day)	195	195	195	195
		Std (day)	0	0	0	0
		E[demand] (year)	71175	71175	71175	71175
		Ch	0,1	0,1	0,1	0,1
		Co	20,00	20,00	20,00	20,00
		Price	$ 30,00	$ 30,00	$ 30,00	$ 30,00
		Target service Level	100,00%	100,00%	100,00%	100,00%
		Lead Time (day)	0	0	0	0
Decision Variables		Q	974,166	1200,000	500,000	974,166
		P	4,996	6,154	2,564	7,000
		Q (rounded)	975,000	**1200,000**	**500,000**	975,000
		P (rounded)	5,000	**6,000**	**2,000**	**7,000**
Policy evaluation	Theoretical	Orders:	73,062	59,313	142,350	52,143
		Avg. Stock:	487,083	600,000	250,000	487,083
		Co	$ 1 461,25	$ 1 186,25	$ 2 847,00	$ 1 042,86
		Ch:	$ 1 461,25	$ 1 800,00	$ 750,00	$ 1 461,25
		Ca:	$ 2 135 250,00	$ 2 135 250,00	$ 2 135 250,00	$ 1 523 874,44
		TC	**$ 2 138 172,50**	**$ 2 138 236,25**	**$ 2 138 847,00**	**$ 1 526 378,55**
	Simulation	Fill rate	100%	100%	100%	72%
		Serv. Level	100%	100%	100%	0%
		Orders:	73,000	61,000	183,000	53,000
		Avg. Stock:	478,750	799,250	498,730	377,450
		Co:	$ 1 460,00	$ 1 220,00	$ 3 660,00	$ 1 060,00
		Ch:	$ 1 462,50	$ 2 397,76	$ 1 496,20	$ 1 048,99
		Ca:	$ 2 135 250,00	$ 2 147 850,00	$ 2 150 250,00	$ 1 550 250,00
		TC:	**$ 2 138 172,50**	**$ 2 151 467,76**	**$ 2 155 406,20**	**$ 1 552 358,99**

Figure 10: optimal and sub-optimal inventory management policies for product “Coffee table book”

The last column presents an interesting case. The order quantity is equal to the EOQ, but the revision interval (i.e., 7) is greater than the optimal one (i.e., 5). As expected, the number of orders decreases (53 vs. 73). Similarly, the average inventory decreases (377.45 vs. 478.75). As a result, the ordering, holding, and acquisition costs reduce. More precisely, this policy leads to an annual total cost that is almost 38% lower than the optimal policy. However, a quick look at the product fill rate and service level reveals that this policy is infeasible. Indeed, operating under this policy the fill rate is only 72%, while the service level is 0%, meaning that the company would experience stockouts in every replenishment cycle.

Teaching tip: after computing the optimal parameters for this product, give the students a table with different values for Q and P (see grid below) and ask them to “guess” the impact on the costs and fill rate (with respect to the values obtained using the optimal parameters) of those parameters. Give them a couple of minutes to fill the grid, and then run some of the cases on the board using the simulator and discuss the results. This exercise will help students develop intuition into the impact of the parameters in the performance of an inventory system.

Parameters		**Costs**			**Filling rate**
Q	**P**	**Ordering**	**Holding**	**Acquisition**	
Value for Q	Value for P	Higher/Lower	Higher/Lower	Higher/Lower	Increases/Decreases

Policy 2: Soup Nazi bubble head

This product has a constant deterministic demand rate. Therefore, the EOQ model can compute the optimal ordering quantity. Figure 11 shows the computations.

INVENTORY POLICIES

				Q=EOQ | P=P*
Parameters		Mean (day)		12
		Std (day)		0
		E[demand] (year)		4380
		Ch		0,12
		Co		20,00
		Price	$	100,00
		Target service level		100,00%
		Lead Time (day)		0
Decision Variables		Q		120,830
		P		10,069
		Q (rounded)		121,000
		P (rounded)		10,000
Policy evaluation	Theoretical	Orders:		36,249
		Avg. Stock:		60,415
		Co	$	724,98
		Ch:	$	724,98
		Ca:	$	438 000,00
		TC	**$**	**439 449,97**
	Simulation	Fill rate		100%
		Serv. Level		100%
		Orders:		37,000
		Avg. Stock:		71,270
		Co:	$	1 850,00
		Ch:	$	855,29
		Ca:	$	445 300,00
		TC:	**$**	**448 005,29**

Figure 11: EOQ calculation and evaluation for product Soup Nazi bubble head

Policy 3: Himalayan walking shoes, King Edward VIII's wedding cake, and Urban Sombrero

These three products are bought from the same supplier (i.e., J. Peterman). Since the supplier applies an administrative fee of $50 for each order, there is incentive to implement a joint replenishment policy for these three products. Students tend to miss this big picture and treat the three products separately. That policy leads to a total (theoretical) cost of $438,853.40. On the other hand, the optimal joint replenishment policy (see [1] for reference) leads to a total (theoretical) cost of $437 496.93 (an improvement of 0.31%). On the simulated day-to-day operation, the improvement is even better. As a matter of fact, the three independent policies result on a total cost of $464,960.49 while the joint policy results in a total cost of $450,077.32 (an improvement of 3.20%). Figure 12 shows the computations. The instructor can find the EOQ computations for the joint replenishment policy in the "`Policy 3 (EOQ computations)`" sheet of the `nineties-memorabilia_analysis.xslx` spreadsheet file.

INVENTORY POLICIES

			Independent Replenishment (Q=EOQ, P=P*)			Joint Replenishment (Q=EOQ, P=P*)		
			Himalayan Walking Shoes	King Edward VIII's cake	Urban Sombrero	Himalayan Walking Shoes	King Edward VIII's cake	Urban Sombrero
Parameters		Mean (day)	3	4	2	3	4	2
		Std (day)	0	0	0	0	0	0
		E[demand] (year)	1095	1460	730	1095	1460	730
		Ch	0,12	0,12	0,12	0,12	0,12	0,12
		Co	65	65	65	31,67	31,67	31,67
		Price	120	95	225	120	95	225
		Target service level	100%	100%	100%	100%	100%	100%
		Lead Time (day)	0	0	0	0	0	0
Decision Variables		Q	99,425	129,031	59,286	66,112	88,149	44,075
		P	33,142	32,258	29,643	22,037	22,037	22,037
		Q (rounded)	100,000	130,000	60,000	67,000	89,000	45,000
		P (rounded)	33,000	32,000	29,000	22,000	22,000	22,000
Policy evaluation	Theoretical	Orders:	11,013	11,315	12,313	16,563	16,563	16,563
		Avg, Stock:	49,713	64,516	29,643	33,056	44,075	22,037
		Co	$ 715,86	$ 735,48	$ 800,36	$ 524,49	$ 524,49	$ 524,49
		Ch:	$ 715,86	$ 735,48	$ 800,36	$ 476,01	$ 502,45	$ 595,01
		Ca:	$ 131 400,00	$ 138 700,00	$ 164 250,00	$ 131 400,00	$ 138 700,00	$ 164 250,00
		TC	**$ 132 831,73**	**$ 140 170,96**	**$ 165 850,72**	**$ 132 400,50**	**$ 139 726,94**	**$ 165 369,50**
		Total	**$**		**438 853,40**	**$**		**437 496,93**
		Gap (independent vs. joint)						**-0,31%**
	Simulation	Fill rate	100%	100%	100%	100%	100%	100%
		Serv. level	100%	100%	100%	100%	100%	100%
		Orders:	12,000	12,000	13,000	17,000	17,000	17,000
		Avg, Stock:	53,210	70,830	33,400	37,120	49,040	25,140
		Co:	$ 780,00	$ 780,00	$ 845,00	$ 538,33	$ 538,33	$ 538,33
		Ch:	$ 766,26	$ 807,43	$ 901,80	$ 534,52	$ 559,04	$ 678,77
		Ca:	$ 143 040,00	$ 146 490,00	$ 170 550,00	$ 135 120,00	$ 142 595,00	$ 168 975,00
		TC:	**$ 144 586,26**	**$ 148 077,43**	**$ 172 296,80**	**$ 136 192,85**	**$ 143 692,37**	**$ 170 192,10**
		Total	**$**		**464 960,49**	**$**		**450 077,32**
		Gap (independent vs. joint)						**-3,20%**

Figure 12: Independent vs. joint replenishment policy for the 3 products supplied by J. Peterman

Policy 4: SP2000 Slicer Miniature

Similar to the other products analyzed up to this point, the SP2000 Slicer Miniature has a constant and deterministic demand rate. On the other hand, contrary to the other products, this one is eligible for quantity discounts. The procedure to compute the EOQ with quantity discounts is detailed in [1]. The computations are displayed in Figure 13. The first 3 columns show the EOQ computations for each of the price intervals. The EOQs found for the first two ($5 and $4.5) are feasible (i.e., the order quantity lays within the corresponding intervals), while the EOQ found for the last one is not. The computations show that the best option is not to order any of the EOQs but the entry quantity for the cheapest price interval (i.e., 1000). By ordering this quantity, we obtain theoretical savings of 27.9% and 15.2% with respect to the two feasible EOQs. The savings are 19.1% and 13.0% in the simulated environment.

Policy 5: Puffy Shirt replica

The last product has a stochastic demand following a normal distribution with a mean of 400 units/day and a standard deviation of 50. Moreover, this is the only product in the scenario with a lead time different than 0 (i.e., 2 days). We will use a continuous review policy. To compute the reorder point, we first need to compute the safety stock (SS). Following the procedure described in [1], we compute the SS as follows:

$$SS = z \times \sigma_u \times \sqrt{d} = 1.2815 \times 50 \times \sqrt{2} = 90.61$$

where z is the z score of the normal distribution based on the targeted service level, σ_u the standard deviation of the demand, and d the lead time of the product. For our case, the SS is 90.61.

INVENTORY POLICIES

			Q=EOQ ($5)	Q=EOQ ($4.5)	Q=EOQ ($3.9)	Q=1000
Parameters		Mean (day)	28	28	28	28
		Std (day)	0	0	0	0
		E[demand] (year)	10220	10220	10220	10220
		Ch	0,2	0,2	0,2	0,2
		Co	20	20	20	20
		Price	$ 5,00	$ 4,50	$ 3,90	$ 3,90
		Target service level	100%	100%	100%	100%
		Lead Time (day)	0	0	0	0
Decision Variables		Q	639,375	673,960	723,949	1000,000
		ROR	0,000	0,000	0,000	0,000
		Q (rounded)	**640,000**	**674,000**	**724,000**	**1000,000**
		ROR (rounded)	**0,000**	**0,000**	**0,000**	**0,000**
Policy evaluation	Theoretical	Orders:	15,984	15,164	14,117	10,220
		Avg. Stock:	319,687	336,980	361,975	500,000
		Co	$ 319,69	$ 303,28	$ 282,34	$ 204,40
		Ch:	$ 319,69	$ 303,28	$ 282,34	$ 390,00
		Ca:	$ 51 100,00	$ 45 990,00	$ 39 858,00	$ 39 858,00
		TC	**$ 51 739,37**	**$ 46 596,56**	**$ 40 422,68**	**$ 40 452,40**
	Simulation	Fill rate	100%	100%	100%	100%
		Serv. level	100%	100%	100%	100%
		Orders:	16,000	16,000	15,000	11,000
		Avg. Stock:	336,140	354,020	380,500	522,880
		Co:	$ 320,00	$ 320,00	$ 300,00	$ 220,00
		Ch:	$ 336,14	$ 318,61	$ 342,45	$ 407,84
		Ca:	$ 51 200,00	$ 48 528,00	$ 48 870,00	$ 42 900,00
		TC:	**$ 51 856,14**	**$ 49 166,61**	**$ 49 512,45**	**$ 43 527,84**

Figure 13: EOQ with discounts computations for product SP2000 Slicer Miniature

Now we can compute the reorder point (ROR) as follows:

$$ROR = u \times d + SS = 400 \times 2 + 90.61 = 890.61 \approx 891$$

were u is the mean of the demand.

To complete the policy we need an order quantity. A good rule of thumb is to use the EOQ. For our specific case, we can compute EOQ as follows:

$$Q = EOQ = \sqrt{\frac{2 \times C_o \times TD}{C_u \ \times C_h}} = \sqrt{\frac{2 \times 150 \times (400 \times 365)}{100 \ \times 0.10}} = 2092{,}84 \approx 2093$$

Note that EOQ is not necessarily the optimal order quantity but it should lead to a high quality policy.

Contrary to products with deterministic demand, it is difficult to analytically evaluate the total cost of a policy. We can, however, rely on the simulation to compare different policies. Figure 14 shows a few of them. The first column shows the results for the policy we constructed above (ROR=988 and Q=282). Note that the results reported in the figure were obtained simulating the operation of the policy only once. Multiple runs will lead to different, but comparable, results.

INVENTORY POLICIES

		Q=EOQ|ROR=ROR*	Q=EOQ|ROR<ROR*	Q=EOQ|ROR>ROR*	Q<<EOQ|ROR=ROR*	Q>>EOQ|ROR=ROR*
Parameters	Mean (day)	400	400	400	400	400
	Std (day)	50	50	50	50	50
	E[demand] (year)	146000	146000	146000	146000	146000
	Ch	0,1	0,1	0,1	0,1	0,1
	Co	150	150	150	150	150
	Price	$ 100,00	$ 100,00	$ 100,00	$ 100,00	$ 100,00
	Ch rate (day)	0,027	0,027	0,027	0,027	0,027
	Target service Level	0,9	0,9	0,9	0,9	0,9
	Lead Time (day)	2	2	2	2	2
Decision Variables	SS	90,62	-	-	90,62	90,62
	ROR	890,62	-	-	890,62	890,62
	Q	2092,84	2092,84	2092,84	150,00	400,00
	SS (rounded)	91,00	-	-	91,00	91,00
	ROR (rounded)	891	**600**	**1300**	891	891
	Q (rounded)	2093,00	2093,00	2093,00	**600,00**	**3000,00**
Simulation	Fill rate	99,97%	98,42%	100,00%	99,98%	99,97%
	Serv. level	98,57%	72,86%	100,00%	99,59%	98,00%
	Orders:	70	70	71	244	50
	Avg. Stock:	2166,62	1915,56	2507,01	953,50	3037,74
	Co:	$ 10 500,00	$ 10 500,00	$ 10 650,00	$ 36 600,00	$ 7 500,00
	Ch:	$ 21 665,62	$ 19 124,03	$ 25 070,10	$ 9 534,60	$ 30 376,85
	Ca:	$ 14 651 000,00	$ 14 651 000,00	$ 14 860 300,00	$ 14 640 000,00	$ 15 000 000,00
	TC:	**$ 14 683 165,62**	**$ 14 680 624,03**	**$ 14 896 020,10**	**$ 14 686 134,60**	**$ 15 037 876,85**

Figure 14: evaluation of different CRPs for product Puffy Shirt replica

Teaching tip: run each policy and show the students that the resulting total cost is comparable but different in every run. Discuss with them the reasons (the simulated demands are different in each run). This will help them understanding a policy evaluation using simulation.

Teaching tip: ask students to build a PRP for this product. Discuss with them how to select a convenient review interval.

References

[1] Y. Adulyasak. Pedagogical Note on Inventory Management. HEC Montréal, 2017.