**PROCESS ANALYSIS METHODS**

**1. INTRODUCTION**

Some of the key information that any entrepreneur or manager wants to know about his or her business is how much can the business produce per unit of time and under what conditions; that is, how much can it produce of a single product or service or a mix of products and services, or how much can it produce given large or small or even a mixture of order sizes. Information on how long it takes to get orders through the manufacturing or service facility is also required to setup service guarantees regarding delivery and due dates; that is, film will be ready in an hour, loan will be approved in three days, and pizza will be delivered to the door in 30 minutes. Information on how much capital and labor are required in order to meet or exceed these guarantees is also necessary for making tactical as well as strategic decisions. HOM: Process Analysis is designed to help the entrepreneur or manager respond to these questions.

Since the types of problems in the real world that involve the analysis of processes vary dramatically from setting to setting, this HOM module has been designed to be very general and flexible. It facilitates the ability to model many different situations. Therefore, the Process Analysis module is also the most complex to populate with the correct information. We therefore recommend that the user follow the Help tutorial the first few times solving Process Analysis problems.

As with all HOM modules, the basic product data is entered through the data spreadsheet. Due to the complexity of this module, a third data entry vehicle has been created. The data pertaining to each product, resource or labor type is stored in a database which is accessed by selecting the name of a product, resource and labor type. There are five steps to entering and analyzing a process. Before starting this task, however, we recommend that the user write down in a logical and sequential list the set of steps required to create the product or service to be analyzed (we shall use the following example often, to bake cookies requires mixing ingredients, spooning cookies onto trays, baking, cooling, packing, paying).

Currently, HOM can be used to model make-to-order systems that create products or services in a linear sequence of steps. Hints for modeling some atypical situations are given in section 13.2.

**2. STEP 1: Select Time Units and Name Products, Workcenters and Labor**

On start-up, Step 1 of the spreadsheet is presented. The directory of products, workcenters, and labor types are entered (and always indicated thereafter). These columns will have default names P1, M1 and L1 until changed. The user also enters time unit for analysis.

HOM permits the modeling of up to 25 different products, up to 30 workcenters and gives you the opportunity to specify up to 10 types (classes) of labor. The idea is that execution of each step in a process requires a workcenter and a labor type. A labor type (or, equivalently, labor class) could be used to denote a specific skill or experience category. For example, some workcenters might require labor with a particular skill (cook, machinist, repair person, cashier, or clerk). A bakery may require decorators, bakers, mechanics, and unskilled labor. Hospitals may need scrub nurses, floor nurses, nurse’s aides, and so on.

Next, we need to specify the time units. There are five data items necessary:

Run/Setup Time Units Choose the units in which setup and run times

will be expressed. Minutes is the default value.

Time Units: Demand Select the time units in which demand will be expressed,

for example, 10 orders per day, 20 cars per month. The default

value is Minutes.

Time Units: Simulation Decide the units of time in which the length of simulation

will be expressed. The default unit is Minutes.

Conversion factors The next two boxes on the right side of the dialog box are

for entering the relationship between the three time units.

**Warning**: Once you go to Step 2 returning is not possible without deletion of all data.

**Warning**: Make sure you have specified all the names of the machines, products and workcenters. *There is a work around which is some what cumbersome*: If you forgot to name a machine do this. Before going back to Step 1 to add entries you can save the current data. Then, go to step1, reload the saved data, then add the name of the machine that you forgot.

**Warning**: If you don’t go back to a step and make edits then those edits will not be reflected in the remaining steps. For example, if you want to change a workcenter, first go to Step 2, change the workcenter parameters then proceed.

**God news**: You can go back and forth between steps 2-5!

**3. STEP 2: ENTER WORKCENTER DATA**

The descriptions of the data items are given below. These have to be entered for every workcenter. The workcenter can be selected from the drop-down box.

**3.1 Number of Identical Equipment**

Enter how many identical machines there are at a workcenter. In a service operation or where no equipment is used, simply make this equal to the number of workers. There will often be just one machine, such as the oven in a bakery. But in some operations, such as ATMs in a bank, there could be several machines. If an operation has infinite capacity (for example, cooling hot cookies prior to packing), make this a large number, say, 100.

**3.2 Maximum Units That Can Be Processed on a Machine without an Additional Setup**

Enter the maximum number of units that can be processed at a time without additional setup. This number indicates either a physical capacity of the machine or a managerial decision on the maximum lot size allowed. We interpret a lot size to be the maximum number of units that can be processed before an additional setup is required. Note that orders cannot be combined for processing in this version of HOM. In addition, we use the term job to indicate a subset of a customer order that is available for processing at a workcenter. The maximum units information is used to schedule jobs at the workcenter.

Assume that we are discussing the oven in a bakery. Assume that the oven can process three units (maximum) at a time. Assume a unit is 1 dozen. Assume that a job waiting to be baked has the priority to be scheduled (i.e., it can be taken up for processing) next.

If the order size of the job is less than 3 units (3 dozen), and if the entire order is at the workcenter, then the job will be processed as a whole, that is, the full order will be baked at one time. An example would be product that has an order size of 1 dozen. If the entire 1 dozen are available for baking (that is, mixed and placed on trays), then the dozen cookies will be loaded.

If the order size of the job exceeds 3 units, and if at least 3 units of this order are available for processing right now, then 3 units will be processed. In other words, the job will be broken up into smaller lots to accommodate the maximum units constraint.

Finally, note that in both cases described above, the scheduling rule will postpone the processing of a job if the lot size can be made as close to the maximum units (i.e., 3) as possible by waiting. The workcenter is programmed to wait for the rest of the order to arrive at the workcenter. This device is an automatic setup time–saving feature built into all the scheduling algorithms. Other setup time–saving features are described belowuse\_setup\_saving\_workcenter. In any case, this version of HOM does not permit the user to combine two or more orders to fill a machine to capacity.

In a process that allows for large order sizes, such as 1,000 circuit boards in one setup, you may want to test the effects on performance of stopping the machine and loading an order of a smaller size (say, 10 circuit boards) in between processing the large order. (This would be feasible only if the product with the smaller order size has higher priority.) This device *could* also be used to facilitate the flow of the product to the next processing step. By specifying a maximum unit of 100, an order of size 1,000 will get broken up into lots of size 100, processed in 100’s and transferred to the next step for processing, without waiting for the entire 1,000 circuit boards to be completed. Remember too that if you have not specified Use Setup Saving Algorithm (see below)use\_setup\_saving\_workcenter, you would have caused the process to incur additional setups each time you break up an order.

**3.3 Default SCV of Setup Time**

Enter the default value of the squared coefficient of the setup time for a machine in this workcenter. The ability to add this measure of uncertainty to the process makes the outcome more realistic.

**Caution!** Once the default value of the setup time has been set, it can only be changed on the recipe specification step (Step 4). For example, if you have set the default SCV of setup time to be 0.3, this value is automatically entered when this workcenter is part of the process recipe of a product. However, after the process recipe has been created, changes in the default SCV of the setup time will *no longer* be automatically reflected in the process recipe. The SCV values will stay unchanged for the recipes created before the change. However, the new default value of the SCV will be used for any new recipes that you create.

The squared coefficient of variation is the variance divided by the square of the mean. In this instance, it refers to the variance divided by the square of the average setup time. The SCV must be nonnegative. The larger the value of the SCV, the greater the variability in the setup time. If the SCV is set to zero, then the setup time will be deterministic (no variability). If it is set to one, it will correspond to exponentially distributed setup time.

The default value for the SCV is initially set equal to 0.3. Please see section 10 on Modeling Variability for details about how HOM uses just the information about the mean and the variance (via the SCV) to fit a distribution. Also note that this method of approximating distributions is intended only to give the user a (rough-and-ready) feel for the effect of variability and is not guaranteed to be precise. Simulations that use analytical and empirical distributions and model single-stage queueing systems can be carried out using the HOM: Waiting Line Management module.

**3.4 Default SCV of Run Time**

Enter the default value of the squared coefficient of the run time for a machine in this workcenter. The ability to add this measure of uncertainty to the process makes the outcome more realistic.

**Caution!** Once the default value of the run time has been set, it can only be changed on the data spreadsheet. For example, if you have set the default SCV of run time to be 0.3, this value is automatically entered when this workcenter is part of the process recipe of a product. However, after the process recipe has been created, changes in the default SCV of the run time will *no longer* be automatically reflected in the process recipe. The SCV values will stay unchanged for the recipes created before the change. However, the new default value of the SCV will be used for any new recipes that you create.

The squared coefficient of variation is the variance divided by the square of the mean. In this instance, it refers to the variance divided by the square of the average run time. The SCV must be nonnegative. The larger the value of the SCV, the greater the variability in the run time. If the SCV is set to zero, then the run time will be deterministic (no variability). If it is set to one, it will correspond to exponentially distributed run times.

The default value for the SCV is initially set equal to 0.3. Please see section 10 on Modeling Variability for details about how HOM uses just the information about the mean and the variance (via the SCV) to fit a distribution. Also note that this method of approximating distributions is intended only to give the user a (rough-and-ready) feel for the effect of variability and is not guaranteed to be precise. Simulations that use analytical and empirical distributions and model single-stage queueing systems can be carried out using the HOM: Waiting Line Management module.

**3.5 Type of Processing**

Three types of processes can be modeled using HOM. The user should select the type of process most appropriate for each workcenter from the list shown below. Please note that customer orders are not combined for processing in this version of HOM. For first-time users of HOM, we recommend you read all of section 3.5 before making your selection on which process type best resembles the situation you are modeling.

One Unit at a Time

In Batches

Continuous Flow

Also, remember that HOM is currently designed to model make-to-order systems that create products or services in a linear sequence of steps. Hints for modeling some atypical situations and machines are given in section 13.2.

***3.5.1 One Unit at a Time (Type of Processing—Workcenter Dialog Box)***

This is the model of a machine that has setup as well as run time. The setup is an internal setup, that is, the machine is idle while the setup is being done. Nothing else can be done on this machine during its setup and run time. This is typical of most machines available in industry. The machine setup is considered to be finished after the maximum of the machine setup time and the labor setup time has elapsed. For example, machine setup time is 6 min. and labor setup time is 10 min. If both these quantities are deterministic, then the setup is completed after max(6,10) = 10 min. If these quantities are the means of a random distribution, then two random variables having the means 6 and 10 (and appropriate SCVs) are first sampled, then the setup time is considered to be finished after a period of time equal to the maximum of these two quantities. The machine is considered to be free after the setup is completed and a duration equal to the machine run time has elapsed. Labor is considered to be free after the setup time for labor plus the run time (if any) for the labor has been completed. The job is completed after the setup has been finished and an additional duration equal to the maximum of the machine and labor run times has elapsed. This is not the only method of modeling the sequence of setup and run time activities, but has been chosen as a simple but reasonable representation of many actual sequences of activities. However, by appropriately specifying the process recipe, many other sequences can be modeled.

The jobs are processed in lot sizes equal to the minimum of the following quantities:

Order size or

Maximum units that can be processed on a machine without an additional setup

As described earlier, the scheduling rule will postpone the processing of a job if the lot size can be made as close to the maximum units as possible by waiting. The workcenter is programmed to wait for the rest of the order to arrive at the workcenter. This device is an automatic setup time–saving feature built into all the scheduling algorithms. Other setup time–saving features are described in section 3.6use\_setup\_saving\_workcenter.

**Caution!** The time to setup essentially becomes the maximum of the machine and labor setup times. The run time for the product becomes the maximum of the machine and labor run times. This could create a dramatic change in the utilization and flow times! For example, even if setup times were deterministic, if the labor setup time (say 10 min.) exceeds that of the machine (6 min.), then the machine will be idle longer than what the user might anticipate. This would in turn increase the machine utilization. This problem becomes heightened when the setup times are random variables. It is useful to remember that the expected value of the setup time in this case no longer equals the maximum of the two expected values, viz., expected value of the setup time and expected value of the run time.

**Solution?** One solution would be to ensure that the labor setup time is smaller (and, if random, much smaller) than the machine setup time. Also see the examples given in section 14.

***3.5.2 In Batches (Type of Processing)***

This is the model of a machine that processes parts in a batch. The run time of such a machine does not depend on the quantity produced in a batch. For example, it may be possible to bake ½, 1, or 1½ dozen cookies in the oven at the same amount of time. In HOM such a machine is modeled as having a setup time but no run time. Therefore, the setup time entered in the process recipe for a product using this workcenter should be the sum of the setup and run time for the product. For example, if it takes 1 minute to setup the oven and 9 minutes to bake, and if the oven is modeled as a batch process, then enter 10 minutes for the setup time and zero for the run time. Also see the example given in section 6product\_recipe.

HOM will ignore (and warn) any run time entered in the process recipe for such workcenters. The machine is considered to be free after the maximum of the machine setup time and the labor setup time has elapsed. (Nothing else can be done using this machine while it is processing.) Labor is considered to be free after the setup time for labor. The job is completed as soon as the machine is free.

The jobs are processed in lot sizes equal to the minimum of the following quantities:

Order size or

Maximum units that can be processed on a machine without an additional setup

As described earlier, the scheduling rule will postpone the processing of a job if the lot size can be made as close to the maximum units as possible by waiting. The workcenter is programmed to wait for the rest of the order to arrive at the workcenter. This device is an automatic setup time–saving feature built into all the scheduling algorithms. Other setup time–saving features are described in section 3.6use\_setup\_saving\_workcenter.

**Caution!** The time to setup essentially becomes the maximum of the machine and labor setup times. The run time for the product becomes the maximum of the machine and labor run times. This could create a dramatic change in the utilization and flow times! For example, even if setup times were deterministic, if the labor setup time (say 10 min.) exceeds that of the machine (6 min.), then the machine will be idle longer than what the user might anticipate. This would in turn increase the machine utilization. This problem becomes heightened when the setup times are random variables. It is useful to remember that the expected value of the setup time in this case no longer equals the maximum of the two expected values, viz., expected value of the setup time and expected value of the run time.

**Solution?** One solution would be to ensure that the labor setup time is smaller (and, if random, much smaller) than the machine setup time. Also see the examples given in section 14.

***3.5.3 Continuous Flow (Type of Processing)***

The third type of machine processing is continuous flow. In this type of process environment, orders pass right behind one another without having to wait for the previous order to finish. In contrast to the one-unit-at-a-time and batch processes, jobs can be loaded one after the other, for example, to emulate a chain broiler. In a bakery, a continuous flow oven would have openings at either end to allow for the entry of cookie trays as soon as the previous one clears the opening. The continuous flow process is modeled in HOM as one that has both a setup time and a run time. The setup time is the time during which the job is loaded onto the machine. The machine cannot be used for loading another job while a job is being loaded. The machine is considered to be free for *loading* another job after the maximum of the machine setup time and the labor setup time has elapsed. Labor is considered to be free after the setup time for labor. Once the setup has been completed, the job requires an additional duration equal to the run time to finish its processing. Please note that the run time (to be entered in the process recipe) for a continuous flow machine is equal to the time for the job to flow from one end of the machine to the other.

The capacity of such a process is determined by the loading time and physical size. The capacity is not determined by the run time. Throughput time, however, is affected by run time. The physical size is modeled in HOM as the input given in the “Maximum Units that can be Processed” edit box. The jobs are processed in lot sizes equal to the minimum of the following quantities:

Order size or

Maximum units that can be processed on a machine without an additional setup

As described earlier, the scheduling rule will postpone the processing of a job if the lot size can be made as close to the maximum units as possible by waiting. The workcenter is programmed to wait for the rest of the order to arrive at the workcenter. This device is an automatic setup time–saving feature built into all the scheduling algorithms. Other setup time–saving features are described in section 3.6use\_setup\_saving\_workcenter.

**Caution!** The time to setup essentially becomes the maximum of the machine and labor setup times. The run time for the product becomes the maximum of the machine and labor run times. This could create a dramatic change in the utilization and flow times! For example, even if setup times were deterministic, if the labor setup time (say 10 min.) exceeds that of the machine (6 min.), then the machine will be idle longer than what the user might anticipate. This would in turn increase the machine utilization. This problem becomes heightened when the setup times are random variables. It is useful to remember that the expected value of the setup time in this case no longer equals the maximum of the two expected values, viz., expected value of the setup time and expected value of the run time.

**Solution?** One solution would be to ensure that the labor setup time is smaller (and, if random, much smaller) than the machine setup time. Also see the examples given in section 14.

**3.6 Use the Setup-Saving Algorithm**

The last entry in this (workcenter capabilities) step is your decision on whether to use a setup saving algorithm. It must be individually selected for each workcenter in which the algorithm is to be used. An important restriction for using the setup saving algorithm in a workcenter is that the number of identical machines must be less than or equal to 20.

This algorithm when activated for a workcenter allows the workcenter operator to look down the queue of waiting jobs and select the order for processing that requires the same setup process as the order just processed at a machine. This would save the time required to reset the machine if a different order type or process step was run in between two similar jobs. When setup saving is activated, the priority of products is ignored. In case a similar setup cannot be performed, the scheduling rule reverts to the product priority–based scheduling rules described under Type of Processing.

In our cookie-making example, assume that choc chip cookies do not require a setup for mixing the ingredients if the previous order had been for a choc chip cookie, but need a five-minute setup to clean up after an order for sugar cookies has been processed. Assume also that the ingredients for choc chip cookies were mixed for the previous order and that the setup-saving algorithm had been involved. Then, five minutes will be saved if, from two waiting orders, one for sugar cookies and the other for choc chip cookies, the choc chip cookies order was taken up first for processing.

**4. STEP 3: ENTER LABOR CLASSIFICATION DATA**

In this step, you get to enter the number of identical workers of this type that are currently available, as well as the default values for the SCVs of the labor setup and run times. Incorporating uncertainty in the labor setup and run times enhances the reality of HOM’s modeling ability. The rules regarding the use and change of default SCVs are identical to the rules for a workcenter’s default SCVs, and are explained under sections 3.3 and 3.4, Default SCV of Setup time and Default SCV of Run Time. The data is entered for every type of labor. The type can be selected using the dropdown box.

**Note!**  Please refer to section 3.5, Type of Processing, for the rules on scheduling labor, as well as the description of the Parameters dialog box given in Section 7, for the tie-breaking rule for deciding which workcenter to attend when there are multiple workcenters operated by the same type of labor. These rules become quite important when labor and (or) workcenter setup and (or) run times are stochastic (i.e., SCV > 0).

**5. STEP 4: ENTER THE PROCESS RECIPE**

After completing the data entry for all labor types, proceed to Specify Process Recipe. The recipe has to be specified for every product.

The demand data is entered in first. This data comprises four items: order size, demand, SCV (squared coefficient of variation), and priority. Detailed descriptions of these data items are given below.

**5.1 Order Size (Enter Product Demand)**

The order size is the number of *units* demanded by a customer for that product. It is fixed for a given product and should be a positive integer. This particular choice of modeling the zero customer-to-customer variation in the number of units demanded is explained below.

HOM is not designed to be an exact Demand replicator since it can handle only 30 types of products. Thus, if the process were the baking of cookies as per customer orders, you should first decide the “units” of an order. Units choices such as: a single cookie, ½ dozen cookies, and 1 dozen cookies all are possible “units” for the analysis. (This choice is not entered any where. It is for you to choose and express everything in terms of the unit.) If we assume that our “unit” is ½ dozen, then the possible products could be

**Product Order Size Cookie Type Description**

A 1 Choc Chip Orders of ½ dozen choc chip cookies

B 2 Choc Chip Orders of 1 dozen choc chip cookies

C 3 Choc Chip Orders of 1 ½ doz. choc chip cookies

D 4 Choc Chip Orders of 2 dozen choc chip cookies

E 1 Sugar Orders of ½ dozen sugar cookies

F 2 Sugar Orders of 1 dozen sugar cookies.

Notice that by specifying four different order sizes for the “same” product, namely choc chip cookies, we have created some degree of variation in the order size. This method allows the user to set different priorities for customers with different order sizes for choc chip cookies.

In other cases, order size could be used to emulate small, medium, and large orders, for example, 10, 100, and 1,000 circuit boards, with the single circuit board being the basic “unit.” Alternate routing, rework, and scrap can also be modeled by creating multiple products, each with a different routing. The next item to be specified is the demand for the product.

**5.2 Demand (Enter Product Demand)**

The *average* demand is entered in this box. Enter the expected number of orders per *time period*. The default value of the demand *time period* is minute; that is, if you enter a value of 20 in column E, row 3, HOM will assume that on the average there are 20 orders per minute for this product.

In our cookie-making example, the demand for product A could be 1 order per minute, that is, 1 order for ½ dozen choc chip cookies per minute. After entering the demand, proceed to the SCV (squared coefficient of variation)scv with regard to demand.

**5.3 Squared Coefficient of Variation (SCV) with Regard to Demand**

The next item describing the demand is the squared coefficient of variation. This gives the user the opportunity to model uncertain demand arrival patterns. The squared coefficient of variation is the variance divided by the square of the mean. In this instance, it refers to the variance divided by the square of the mean time between two customer arrival epochs (i.e., the SCV of the inter-arrival time between orders; please see note below). The SCV must be nonnegative. The larger the value of the SCV, the greater the variability in the order arrival process. If the SCV is set to zero, then the associated interarrival times will be deterministic (no variability). If it is set to one, it will correspond to exponentially distributed times. Values greater than one will correspond to larger variability.

The default value for the SCV is 0.3. In our cookie-making example, assume that the demand for product A is 1 order per minute. Then if the SCV for product A were zero, an order for ½ dozen choc chip cookies will arrive every minute.

Notice that an SCV of 0.09 represents substantial variability in many instances. This value of SCV implies that the standard deviation of the time between arrivals is 30 percent (0.3) of the value of the mean. In our example, this would imply a standard deviation of 0.3 minutes when the mean time between arrivals is one minute, that is, the range given by the mean +/- 3 standard deviations would be as large as [0.7, 1.3 minutes]. An SCV of 0.01 can be used to emulate moderate variability.

Please see the section 10 on Modeling Variability for details about how HOM uses just the information about the mean and the variance (via the SCV) to fit a distribution. Also note that this method of approximating distributions is intended only to give the user a (rough-and-ready) feel for the effect of variability and is not guaranteed to be precise. Simulations that use analytical and empirical distributions to model single-stage queueing systems can be carried out using the HOM: Waiting Line Management module.

After entering the SCV, please proceed to specifying the priority of the product.

**Note**: Using a single product, we can only model variability in the arrival pattern (time between arrivals) but not in the volume of demand. Order size variation can be handled by using multiple products with different order sizes and carrying out sensitivity analysis with regard to the product mix as well as order size.

**5.4 Priority of Product (Enter Product Demand)**

The priority of a product is a device that allows the program to break ties when determining which product to schedule for processing next. The value for priority can range from 1 to 100 and should be an integer. The number one (1) represents the highest priority and the number 100 the lowest priority.

In our cookie example, we may decide that since orders for two dozen choc chip cookies (product D) are the most profitable, they should have precedence over the other products. So we may assign product D a priority of one (1). Orders for ½ dozen sugar cookies (product E) could be the least profitable, and thus assigned a priority of 10. The priorities of all other products could take values in between 1 and 10. The computer assumes that the ranking is ordinal and disregards the relative difference between the priority values.

**5.5. SPECIFY PROCESS RECIPE**

To this point you have completed a list of all the resources that are available to your operating system (workcenters and labor), and the demand characteristics and priorities for all your products. All that remains to be done is to specify the process recipes that will allow you to use your resources in a unique step-by-step procedure to create your products or services. The current version of HOM allows you only to create products or services that are produced sequentially. Thus HOM cannot be used to model the simultaneous building of an engine and the body of a car, putting them in inventory, and drawing upon them later to build a car. The recipe must be stated in the logical order of processing, that is, Mix --> Bake --> Cool --> Pack --> Pay would be the logical sequence for baking cookies in our bakery example. In the same recipe, multiple visits to the same workcenter are allowed.

Begin the data entry with the name of the process step. Enter the first step of the process; in the bakery example, this might be mixing. Moving to the next column, that is, D, in the same row, enter the name of the workcenter that accomplishes this task. There could be several workcenters that could do the mixing, for example, hand mixing or automatic mixing, but you have to specify a unique workcenter that will accomplish this task. If you type in the name of a workcenter that is not in the list shown in the second column (under Workcenters), a dialog box will open asking, “Workcenter with the given name does not exist.” Re-enter the correct name of the workcenter. The only way to add a new workcenter would be to go back to step 1 – before going back save the data.

In the next two columns enter the average run time and setup time for a machine in this workcenter for carrying out this process step. Repeat the procedure for the name of the labor type that will perform this step of the process recipe, and enter the average setup and run time for the labor as well. The manner in which HOM uses the setup and run time information to carry out the simulation is explained in section 3.5 on the Type of Processing. It has been assumed that the setup and run times will be given in minutes. This unit of time can be changed (only) globally by selecting the appropriate entry in the Step 1.

It is often the case that the labor time is less than the workcenter time. As an example, worker presence may be necessary to setup the oven, but once having set it up, the labor might not be necessary while the oven is baking.

Please note that a workcenter that performs a batch process has only a setup time. Any entry in the run time columns will be ignored (after warning the user). For example, the oven in our example can bake ½, 1, or 1½ dozen cookies in 10 minutes. The loading time for labor is 1 minute. Model this as a batch process, with the maximum number of units that can be processed without requiring an additional setup equal to three, workcenter setup time equal to 10, and labor setup time equal to one.

In a workcenter that performs a continuous flow operation, the setup time corresponds to the time for loading the job, whereas the run time corresponds to the time to flow from one end to the other. The run time for a job on a continuous flow machine therefore does *not* depend on the size of the job. For example, in the bakery example, a bread line requires a flow time of 48 minutes (from end to end), and 2 minutes are required to load a tray of dough (to be baked). The tray can hold up to 4 loaves. The unit for the product is a loaf of bread. Model this as a continuous flow process, with maximum units that can be processed without requiring an additional setup equal to 4, workcenter setup time and labor setup time equal to 2 min., and run time for the workcenter equal to 48 min. No labor run time is permitted for continuous flow processes.

The last four columns show the default values of the SCVs of setup and run times entered previously under workcenter and labor data. If you so desire, these values can be individually edited. Note that the SCV values cannot be changed globally in a simple manner. Please consult sections 4.3 and 4.4 on Default SCV of Setup Time and Default SCV of Run Time. The above procedure is followed for each step in the process recipe. Once all process recipes have been completed, move on to Specify Model.

**Hint!** You cannot insert a row in the process recipe. Therefore, if you forget to include a step in the recipe, you must use the cut, copy, and paste spreadsheet operations to create an empty row for entering the forgotten step.

**Now you are ready to carry out the calculations!**

**6. STEP 5: SPECIFY MODEL**

Model specification requires the user to make a set of decisions on operating procedures and units for measuring time. The items are as follows. Details are given after the table.

**Item Description**

Labor Assignment Rule Specify the tie-breaking rule for deciding which workcenter

to attend.

Use Setup Saving Algorithm Force the scheduler to run together jobs requiring the same

setup process.

Length of Simulation Enter the length of simulation.

Computations Specify whether you wish only the theoretical utilization

to be computed and reported or whether the simulation is to

be run and reported.

Trace Workcenters Choose the workcenters whose queues will be traced

during the simulation and plotted at the end of a

successful run.

Select Products for Calculations The last task in problem specification is to choose the

set of products you want the process to produce.

**6.1 Labor Assignment Rule and Setup Saving Rule**

Labor Assignment Rule Specify the tie-breaking rule for deciding which workcenter

to attend in case labor is free and there are two (or more)

workcenters that require the use of the same type of labor.

The choices are to attend to either the workcenter with

the longest queue or the workcenter with the greater

(theoretically computed) utilization.

**6.2 Length of Simulation and Type of Computations**

Length of Simulation Enter the length of simulation. If the unit of simulation was a week, and you wish to simulate 50 weeks demand, enter the number 50.

The thumb rule is that the length should be sufficiently long such that the simulated values such as the quantities produced per unit time, flow times, queue lengths etc become “stable”.

Computations Specify whether you wish only the theoretical utilization to be computed and reported or whether the simulation is to be run and reported.

The theoretical utilization numbers could be used to make adjustments prior to running a long simulation. HOM will not carry out a simulation if the utilization of any resource (workcenter or labor) exceeds 100 percent. In that case, the theoretical utilization values will be useful in determining

the cause for the overutilization.

**6.3 Tracing Queues at Workcenters (Parameters Dialog Box)**

Trace Workcenters Choose the workcenters whose queues will be traced during the simulation and plotted at the end of a successful run.

This is a useful device for understanding the buildup and reduction of work at key workcenters. HOM has the capability of plotting up to 2,000 points on a graph. Thus, the trace will be for the last 2,000 points if the number of values exceeds 2,000. This is quite likely to happen during a long simulation.

**6.4 Select Products for Calculations (Parameters Dialog Box)**

Choose the set of products that you wish to produce. This option allows the user to carry out sensitivity analysis by running the simulation with a variety of product mixes. The names of the products are selected by clicking on them using the mouse.

**7. CALCULATE**

In order to run a previously specified problem, you need only to click on the Calculate button. Simulation is a very CPU-intensive process. The run time could be several minutes. We recommend that you save the data before commencing the run.

We also recommend that the run time be gradually increased (after starting out with a small value for the length of simulation) until you are satisfied with the results. Long simulations also produce long trace files. If the available disk space is inadequate, the simulation will be terminated with a suitable error message or NO OUTPUT! In that case reduce the length of the simulation and do not trace any workcenter.

**8. RESULTS**

After a successful run, the results will be displayed. If only the theoretical utilization is computed, the results will not include any simulated values. The results include

1. Two bar graphs, one each for workcenters and labor types, showing the theoretical versus simulated values of utilization (%).

2. A table of the theoretical versus the simulated values of resource (workcenter as well as labor) utilization (%).

3. For each workcenter, the average number of jobs in queue, the average number of units at the workcenter, the maximum number of units at the workcenter, and the unavoidable delay due to waiting for labor or material are shown.

A job could be either a part or the whole of a customer order. Number of units at a workcenter is the number of units in all jobs at a workcenter (whether in queue or under process). In the bakery example, the unit was ½ dozen cookies. The number of units is then expressed as the total number of ½ dozen cookies either waiting to be processed or under process at a workcenter. The unavoidable delay is the delay due to unavailability of labor or the delay due to waiting for material to arrive. The latter is caused by the program’s attempt to make the lot size as close as possible to the maximum number of units that can be processed at the workcenter without an additional setup; see Type of Processing.

4. For each product, the average and standard deviation of the flow time are given. The time unit for reporting the flow time is the unit chosen by the user for the setup and run time. The quantity produced of each product during the length of the simulation is also reported in number of units (i.e., number of ½ dozen cookies in our bakery example).

Notice that orders can be split and processed in lots due to the fact that the maximum number of units that can be processed at a time at some workcenter may be smaller than the order size for the product. Therefore, the order might not be delivered in one complete whole (in the order size) at a time to the customer. The order may be delivered in one or more lots. In order to cope with this phenomenon, HOM computes flow time based on weighting the flow time by the quantity produced. For example, if the customer orders two dozen cookies, one dozen of which are delivered in 26 minutes and the rest in 36 minutes, then the weighted average flow time is 30 minutes.

5. The flow time distribution is reported for each product. The distribution is given as a table containing 10 class interval values and the fraction of units of the product that experienced a flow time less than or equal to the right endpoint of the class interval. This is a cumulative distribution of the flow time. If the user desires, these values can be copied using the EDIT --> Copy command and pasted for further analysis or plotting in a commercial spreadsheet program. See section 10 on exporting and importing.

**9. MODELING VARIABILITY**

SCV stands for the squared coefficient of variation, which is defined as the ratio of the variance to the square of the mean. The SCV must be nonnegative. The larger the value of the SCV, the greater the variability in processing (or interarrival) times. If the SCV is set to zero, then the associated processing times will be deterministic (no variability). If it is set to one, it will correspond to exponential processing times. Values greater than one will correspond to larger variability. The value of the SCV is used by the program to sample from an appropriate mixture of a constant (deterministic) and an exponential distribution.

**10. Export and Import of Data as well as Results**

HOM has the ability to export final and interim results to a csv file. When you go back to Step 1 or when you exit the program it asks whether you want to save the data. The reload button can be used to reload previously saved results.

**11. Summary of Processing Sequence**

1. Based on the input data, the program first creates an output showing the data input by the user.

2. The program calculates resource and labor utilization *assuming* that the lot size is equal to the minimum of the order size and the maximum size allowed by the resource. This value is simply an approximation, and simulated utilization values are later displayed. However, for practical purposes, the utilization computed in this fashion should be very close to the simulated utilization.

3. The program keeps track of the time during which resources are idling due to lack of materials (i.e., when a workcenter postpones the processing of a job, preferring to wait for more of the same order to arrive at the workcenter to better utilize the machine) or waiting for labor. The percentage of "unavoidable" idle time is reported. This time can be avoided by adding more labor or increasing the maximum size permitted (see section 3.2).

4. Customer orders arrive as per the random interarrival times specified by the user. They are given an order number (i.e., an ID). Each order has an order size equal to the value input for that type of product.

5. Each resource has a queue associated with it, and jobs wait in the queue until they are taken up for processing.

6. The program keeps track of customer orders as they flow through the shop. In particular, information on the number of units in the order that are still at previous processing steps is maintained. The program also keeps track of the time taken to flow through the shop.

7. When a resource falls idle, or when labor becomes available, or when an order either arrives or departs from a processing step, the program checks to see whether another job can be scheduled.

7.1 The scheduling rules when a labor type becomes free:

7.1.1 If the option of attending the longest queue has been chosen, then the workcenter queues are scanned in the order of decreasing number of waiting jobs— that is, highest priority is given to try and schedule the resource with the largest number of waiting jobs.

7.1.2 If the option of attending the workcenter with the maximum utilization first has been chosen, then the workcenters are scanned in the descending order of machine utilization. The theoretical value of the utilization is used for this purpose.

7.2 In order to schedule a job, a machine must be available, the correct type of labor must be available, and the job must be of sufficient size. Jobs are scanned in decreasing value of priority (priority is set at the level of products). Jobs further along the processing sequence are given higher priority (compared to jobs of the same product but lower down the processing sequence).

7.2.1 If setup saving has not been activated for this workcenter, then a job is taken up for processing if it satisfies the priority rules, a machine of the right type is available, and the right type of labor is available, and (i) the number of units in the job is greater than or equal to the lot size for the resource or (ii) if the order size is less than the lot size, the entire order is available for processing.

7.2.2 Else, the workcenter operator looks down the queue of waiting jobs and selects the order for processing that requires the same setup process as the order just processed at a machine. This would save the time required to reset the machine if a different order type or process step was run in between two similar jobs. When setup saving is activated, the priority of products is ignored. In case a similar setup cannot be performed, the scheduling rule reverts to the product priority–based scheduling rules described above.

7.3 If a job does not satisfy any of the rules, then it waits in the queue and other jobs are scanned for scheduling.

7.4 The computation of processing times is as follows: (i) If the machine is either of the One Unit at a Time or Batch type of processing, then (a) the job finishes its processing after = maximum(machine's setup time, labor setup time) + maximum(machine run time, labor run time); (b) the machine is set free after = maximum(machine's setup time, labor setup time) + machine run time; (c) the labor becomes free after = labor setup time + labor run time. There is a slight inconsistency between (a) (b) and (c), but this usually does not create too much distortion from our experience in running the simulations. (ii) If the machine is of the continuous flow type, then (a) the job finishes its processing after = maximum(machine's setup time, labor setup time) + machine run time; (b) the machine is set free after = maximum(machine's setup time, labor setup time) + machine run time; the labor becomes free after = labor setup time. Note that the job has to flow through the machine for a fixed time = the run time of the machine. The machine is *busy* while the job is being loaded. The labor is *busy* when doing the loading.

8. Once a job has finished its processing it departs. It may depart from the system or go to the next step of processing, as the case might be. Jobs progress through the shop and leave as soon as all processing steps are complete. Customer orders do not wait to be completed to leave the system. For example, an order of size = 40 may be split into 2 lots according to the processing rules. Each lot will depart on its own. The flow time reported is a weighted average per unit—that is, it equals the flow time weighted by the number of units in the job.

9. Once a processing step has been completed, machine and labor are set free (Freemachine and Freelabor).

10. At the end of simulation, reports are generated by using the statistics generated during the simulation.

**12. HINTS and Common Mistakes for This Application**

**12.1 Error Messages**

If the input data was entered incorrectly, or the data integrity rules were not followed, the simulation will not be run. An appropriate error message will be given. Follow the directions in the message to correct the situation.

If the simulation aborts for some reason, please note down the error message that appears on the screen. Usually the cause for the program to abort execution is that some resource(s) or labor category(ies) are being heavily utilized or due to machine interference (due to the same labor type attending more than one workcenter). This (in extreme cases) causes the data to exceed the allowed maximum size (dimension) in the simulation program.

If the program stops execution (after clearing all data integrity checks), it will give a report on the utilization. If you see a resource or labor category over 100 percent utilization, add more resources, or change the product routing.

If the simulated values of utilization do not correspond to the theoretical values, or to your expectations, then the problem could be due either to machine interference (see example in section 14) or to the fact that the setup time for the machine is modeled as the maximum of the setup time for the labor and the setup time for the machine. Please see modeling hints given below.

**12.2 Modeling Hints**

Alternate Routes: A method for modeling alternate routes for the same product is to create two or more products with changes in the processing steps. For example, there are both manual and CNC drilling machines in the shop. There is only a single product with demand = 200/day. You have chosen to do a particular step (say #15) on the CNC drilling machine though it can be also done on the manual drill. The program aborts and shows that the CNC drilling machine's utilization is 150 percent whereas the manual drill has a utilization of 15 percent. You can correct this imbalance by creating two products: each, say, with demand = 100/day (total = original demand), but one of the products uses the CNC whereas the other uses the manual drill to do this step (#15).

Order Size: By specifying three different order sizes for the “same” product, we can create some degree of variation in the order size. This method allows the user to set different priorities for customers with different order sizes. In other cases, order size could be used to emulate small, medium, and large orders, for example, 10, 100, and 1,000 circuit boards, with the single circuit board being the basic “unit.” The next item to be specified is the demand for the product. Alternate routing, rework, and scrap can also be modeled by creating multiple products, each with a different routing.

Priority Queues: The hint given above can be used to model a single or multistage priority queueing system. Please see discussions given below.

Setup Time: The time to setup essentially becomes the maximum of the machine and labor setup times. The run time for the product becomes the maximum of the machine and labor run times. This could create a dramatic change in the utilization and flow times! For example, even if setup times were deterministic, if the labor setup time (say 10 min.) exceeds that of the machine (6 min.), then the machine will be idle longer than what the user might anticipate. This would in turn increase the machine utilization. This problem becomes heightened when the setup times are random variables. It is useful to remember that the expected value of the setup time in this case no longer equals the maximum of the two expected values, viz., expected value of the setup time and expected value of the run time. One solution would be to ensure that the labor setup time is smaller (and, if random, much smaller) than the machine setup time. Also see the examples given in section 14.

Continuous Flow: Continuous flow processing can be mimicked by specifying a large number of “one at a time” machines. For an example, see the “cool” workcenter given in section 14.1.

Modeling Atypical Machines: Some machines exist that do not conform to the previous definitions (section 4) and that cannot be explicitly modeled in this version of HOM. One such machine is a batch machine that allows for interruption and replacement. Consider a pizza oven big enough for six pies. Clearly, once the oven reaches its cooking temperature, pies can be added at any time as long as there is a place to put them. Another such machine is one that makes its own inventory. A coffee brewer makes a batch of coffee (i.e., 8–12 cups) that is then poured to fill sequential orders until the reservoir (inventory) is consumed and another pot needs to be brewed.

Both of these machines can be “approximately” modeled in HOM by utilizing *N* one-at-a-time machines where *N* is the physical capacity of the current batch machine being modeled. An 8-cup automatic coffee pot takes 8 minutes to load and brew. The ½ minute to pour each cup would be approximated by 8 one- at-a-time machines each with a setup time of one minute and a run time of ½ minute. Similar assumptions would be made about setup labor. A six-pie pizza oven that takes 6 minutes to load and 18 minutes to bake could be approximated by 6 one-at-a-time machines with a setup time of 1 minute and run times of 18 minutes. The accuracy of these assumptions depends on the performance measure in use. They will give better results for utilization than for flow times. Modeling processes is an art form that is improved with experimentation and ingenuity.

**14. EXAMPLES**

**Note:** The results will change from run to run because of the way random numbers are generated within HOM.

**14.1 Cookie-Making Example (Cookie1doz.xls)**

We model a *deterministic* process for making custom-baked cookies; that is, all SCVs are zero. We assume that

Number of products : 1

Order size : 1 dozen

Demand per day : 44

Workcenters :

**Name Type Number of Machines Maximum Number of**

**Dozen Cookies That**

**Can Be Processed Without Additional**

**Setup**

MIXSPOON One at a time 1 3

OVEN Batch 1 1

COOL One at a time 100 1

PACK One at a time 1 1

PAY Batch 1 100

Labor :

**Name Number of Workers**

Baker 1

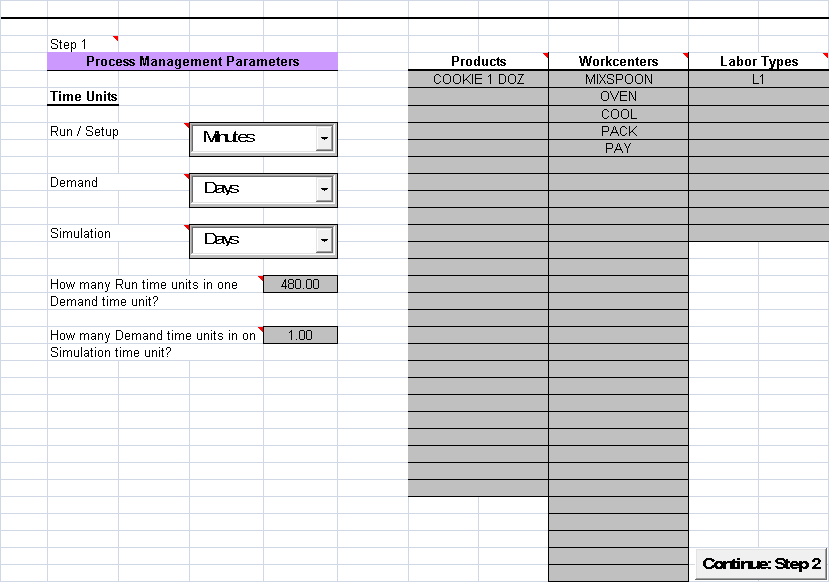
Helper 1

Dummy 100

Process Recipe :

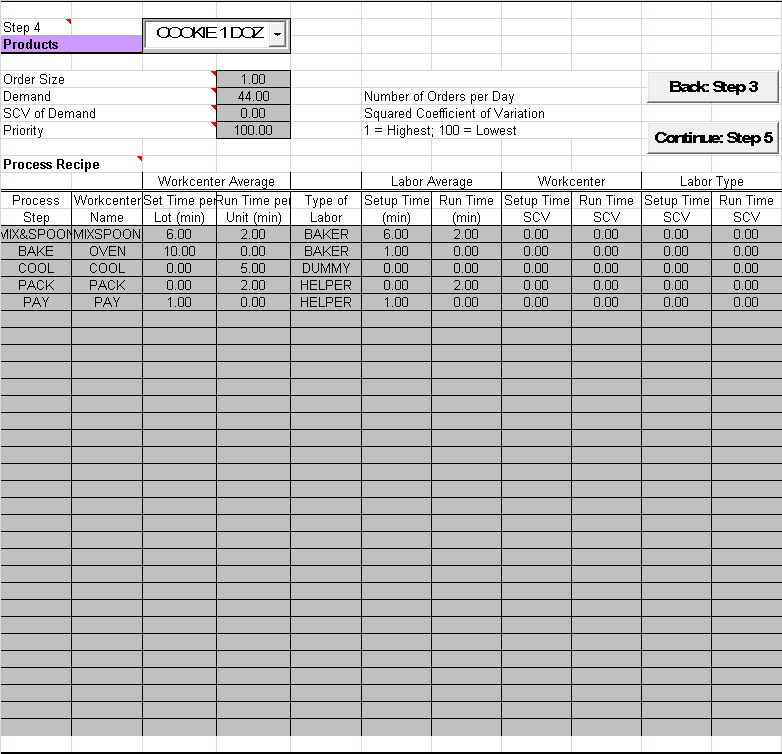
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **Wrkcntr Avg.** | |  | **Labor Avg.** | |
| **Process** | **Name of** | **Setup Time/Lot** | **Run Time/Unit** | **Type of** | **Setup Time** | **Run Time** |
| **Step** | **Workcenter** | **(MIN)** | **(MIN)** | **Labor** | **(MIN)** | **(MIN)** |
|  |  |  |  |  |  |  |
| MIX&SPOON | MIXSPOON | 6 | 2 | Baker | 6 | 2 |
| BAKE | OVEN | 10 | 0 | Baker | 1 | 0 |
| COOL | COOL | 0 | 5 | Dummy | 0 | 0 |
| PACK | PACK | 0 | 2 | Helper | 0 | 2 |
| PAY | PAY | 1 | 0 | Helper | 1 | 0 |

The Step 1 inputs are as follows:

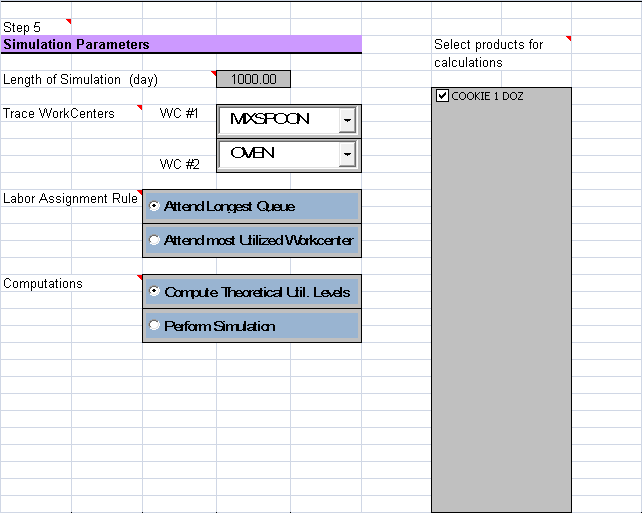


Notice the choice of time units as well as the conversion factors.

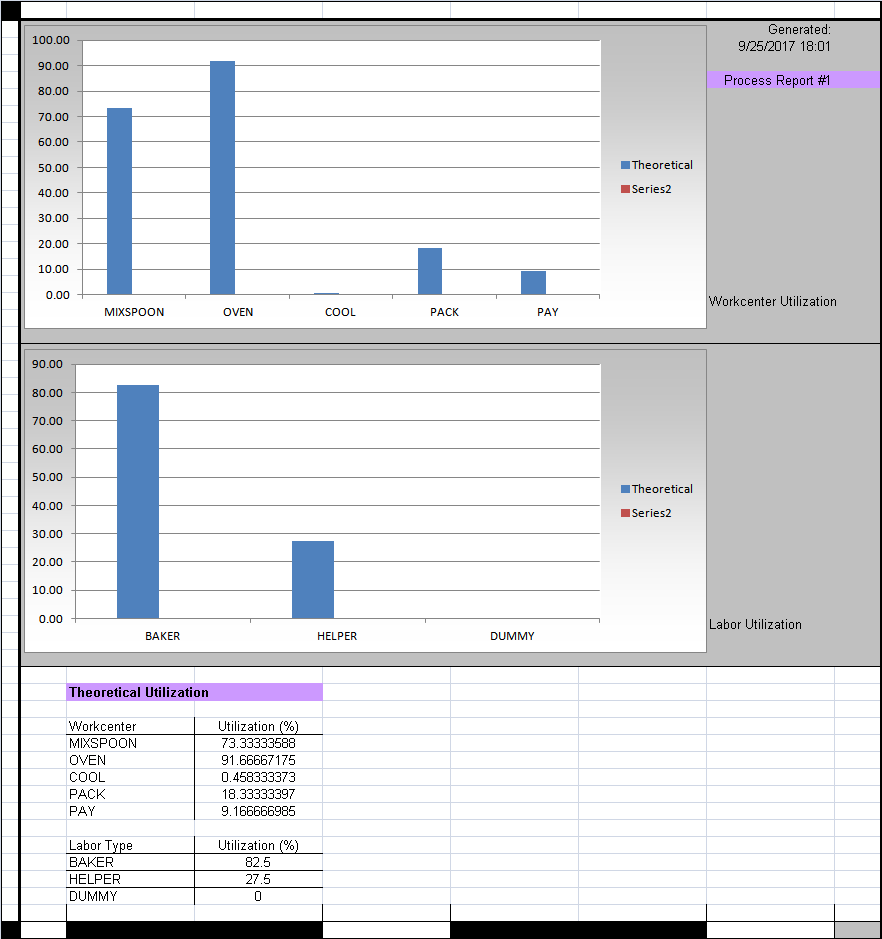
The Step 2 and 3 inputs are entered using the given data. These are not shown. The data after entering in Step 4 are shown below



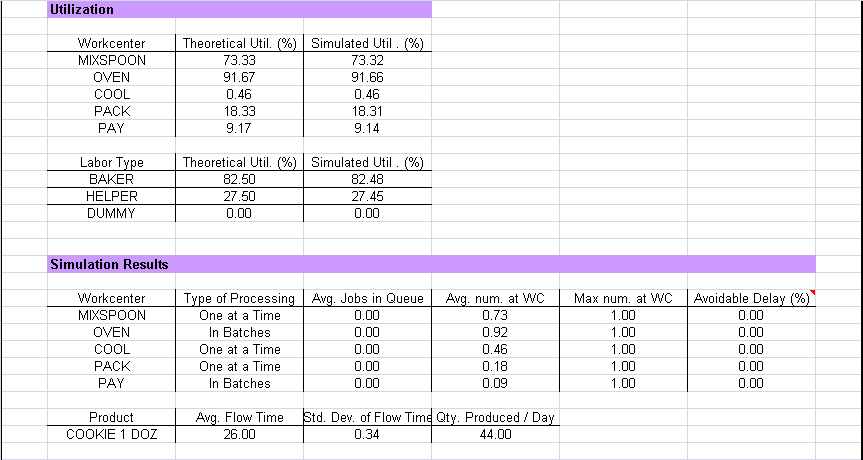
In the calculation step, the product was selected, simulation for 1000 units was specified, longest queue choice was selected and the first two workcenters were traced:



**First the theoretical utilization was computed as shown below:**



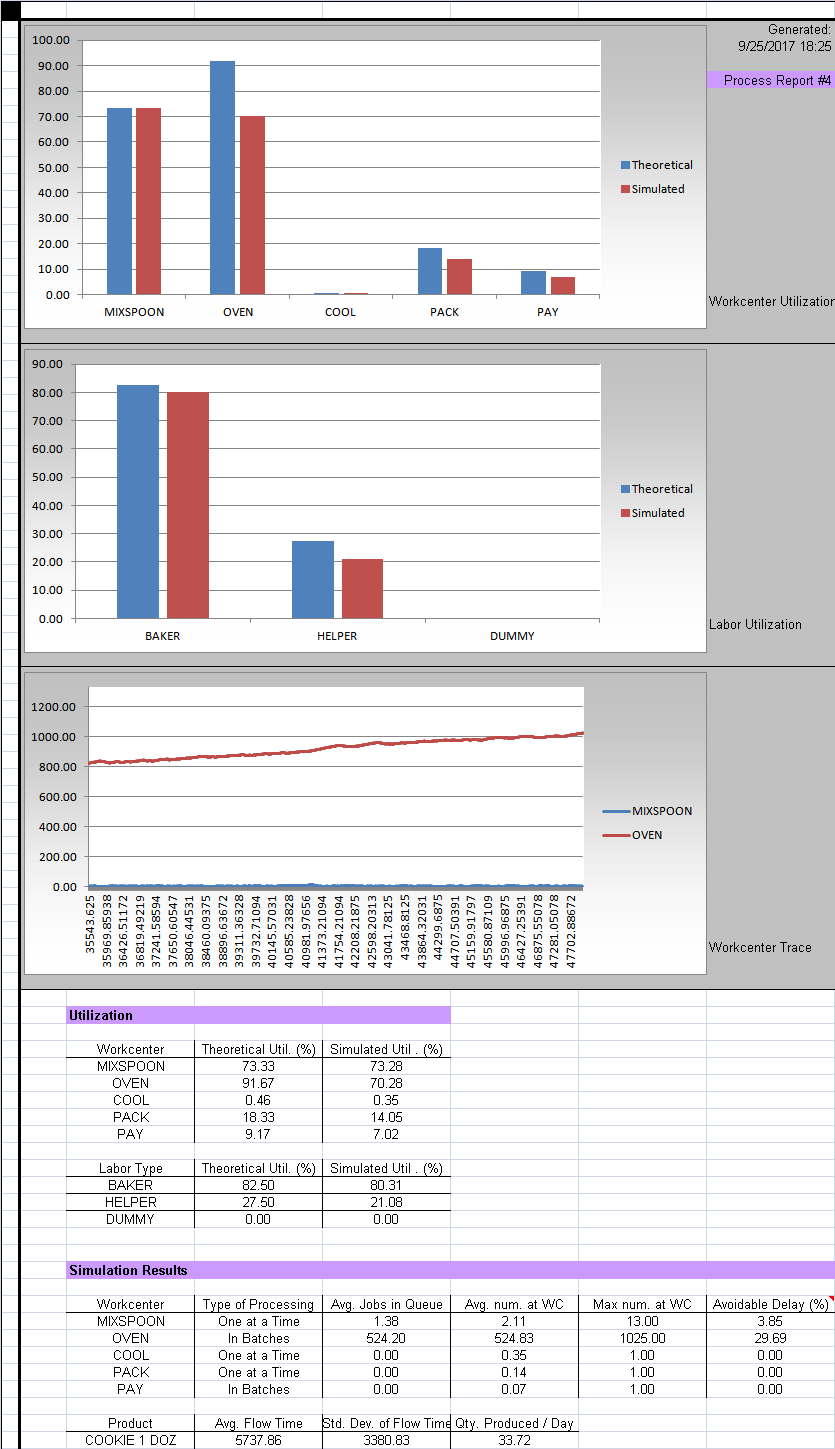
The summary result of the simulation for 1000 days is shown below. (The Perform Simulation radio button should be selected instead of Compute Theoretical Utilization Levels.)



**14.2 Cookie-Making Example—Variations**

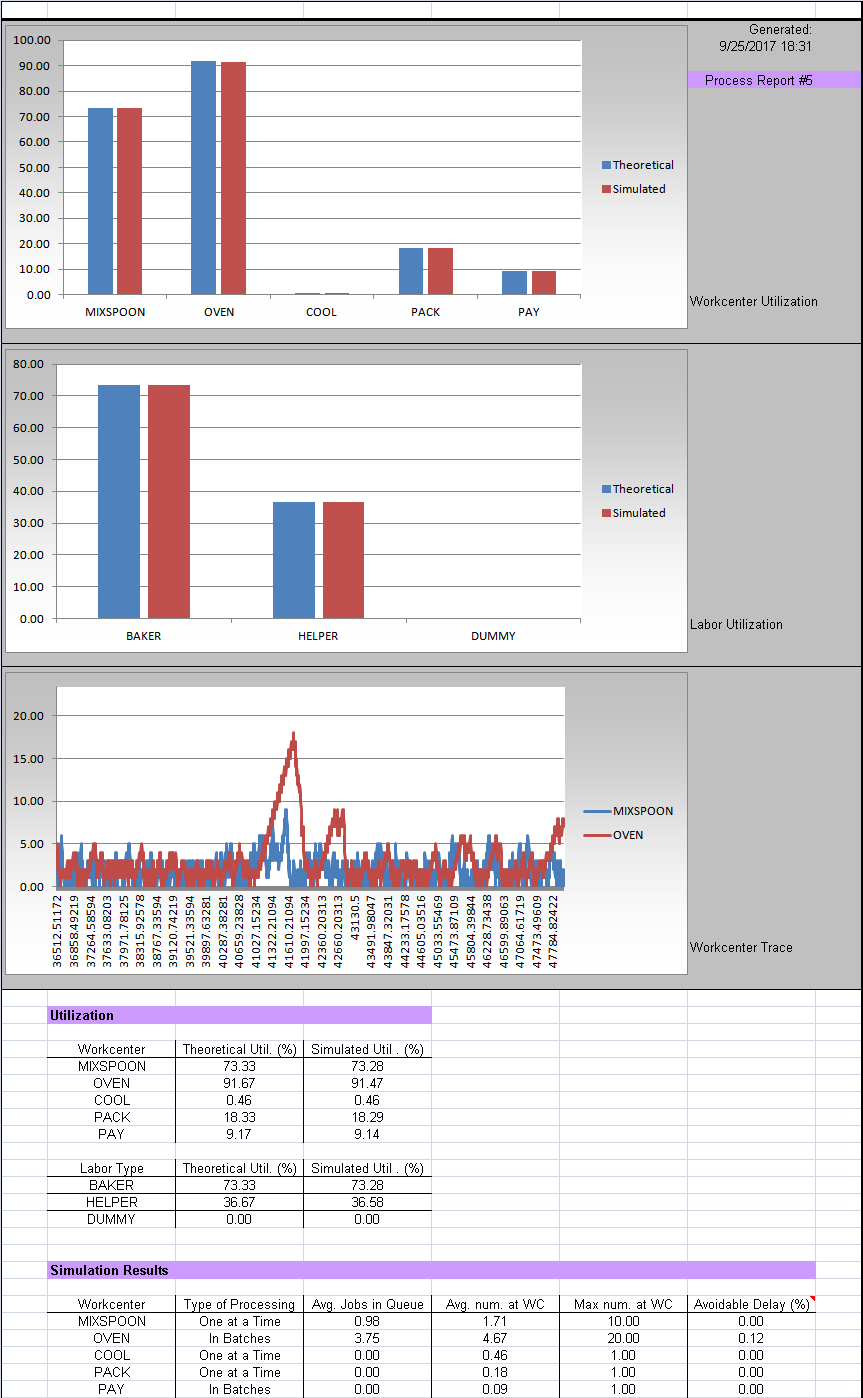
The following variations of this example can be created:

1. Instead of the orders arriving in a deterministic manner, the SCV of demand has been changed to 1 in the file *Cookie1dozRandom.xls*. You will notice that the queue at the oven will be built up and that the system is **unstable**! \*\* warning \*\* don’t simulate for more days!! Ideally simulate for 200 days. Else, computer can freeze.

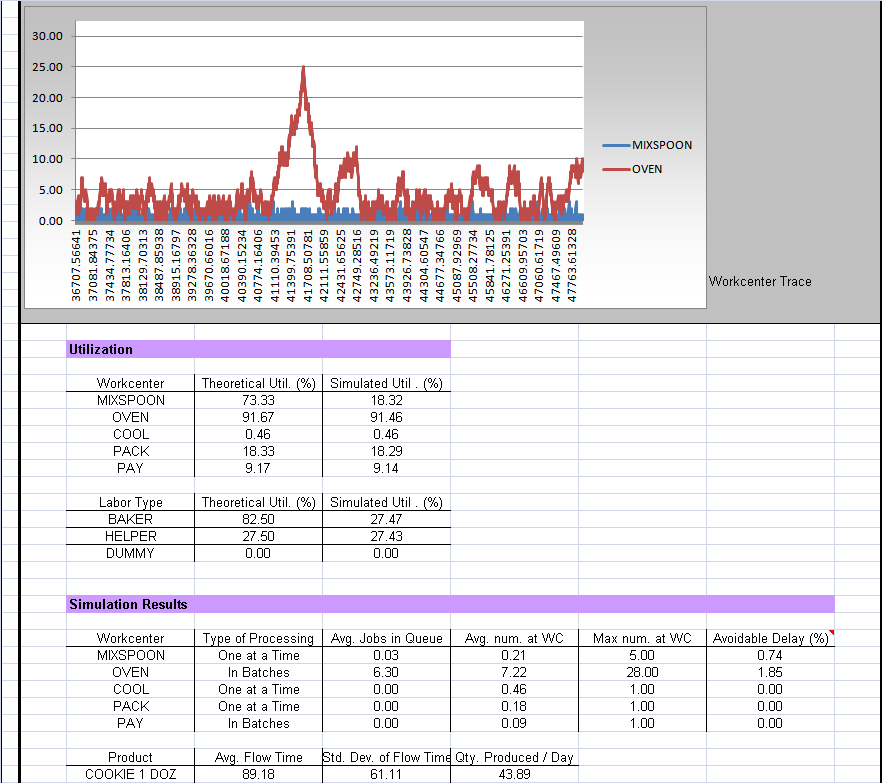


This build-up is due to a phenomenon called *machine interference*. Notice that the unavoidable delay at the OVEN is in excess of 29 percent. The problem arises because the Baker is also heavily utilized, and therefore cannot be in two places (the MIXSPOON and the OVEN) at the same time.

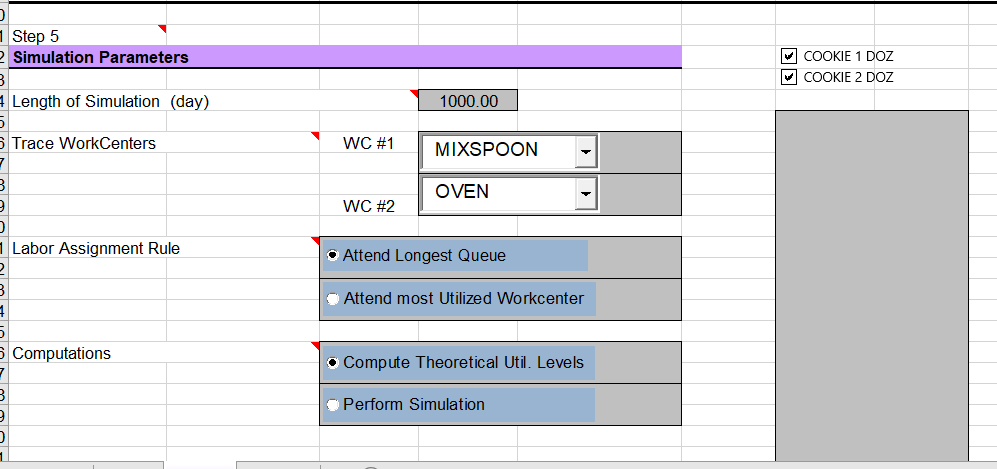
A simple solution would be to assign the Baker’s work to the Helper so far as the loading of the oven is concerned. Try it. See *Cookie1DozReassign.xls*. Now the system becomes stable. The traces are given below.



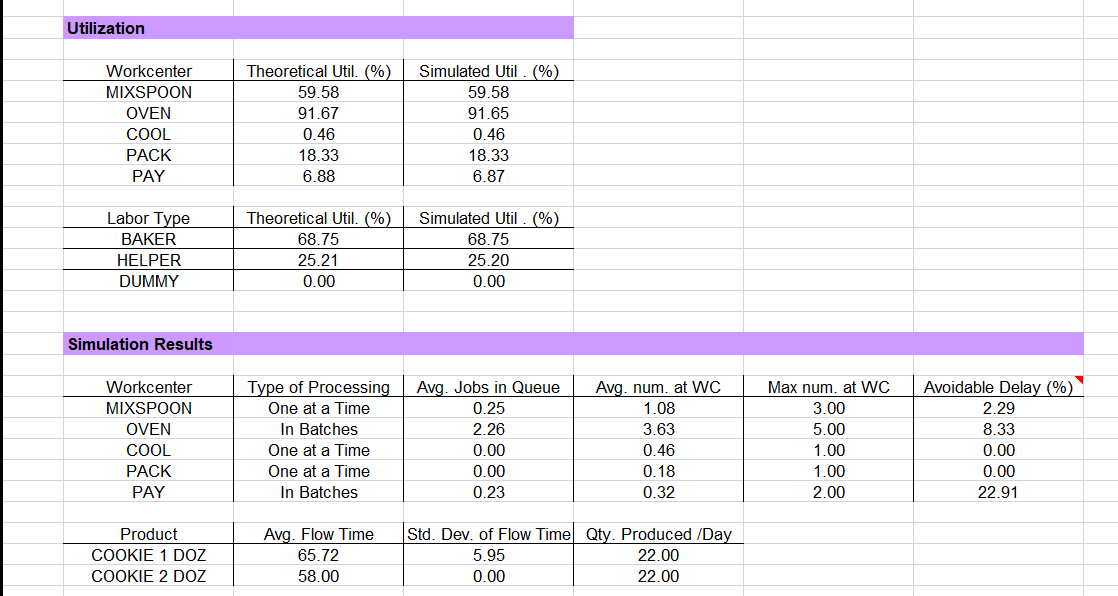
Another simple solution is to activate the setup saving-algorithm at MIXSPOON. (See *Cookie1dozRandomSetupSave.xls*) This essentially eliminates the setup time at this workcenter. This action eliminates the setup time at MIX&SPOON and thus frees up the Baker to handle both operations without causing unnecessary idle time. (You will also notice that the simulated and the theoretical values of the utilization of MIXSPOON are very different from one another, 18.32 percent against 73.33 percent. This is due to setup savings.) The traces are shown below.



2. A second modeling variation has been created in *Cookie1doz2doz.xls*. In this example, a second product has been introduced. This product models orders for two dozen cookies. The process recipe is identical to that used to make orders of a dozen cookies. The user could experiment the effect of product mix variation (by changing the demand values) as well as priority for the products (the priority for both types of orders has been set equal to 100 presently). Initially, the mix is set to 50-50. To simulate, select both products as shown below:



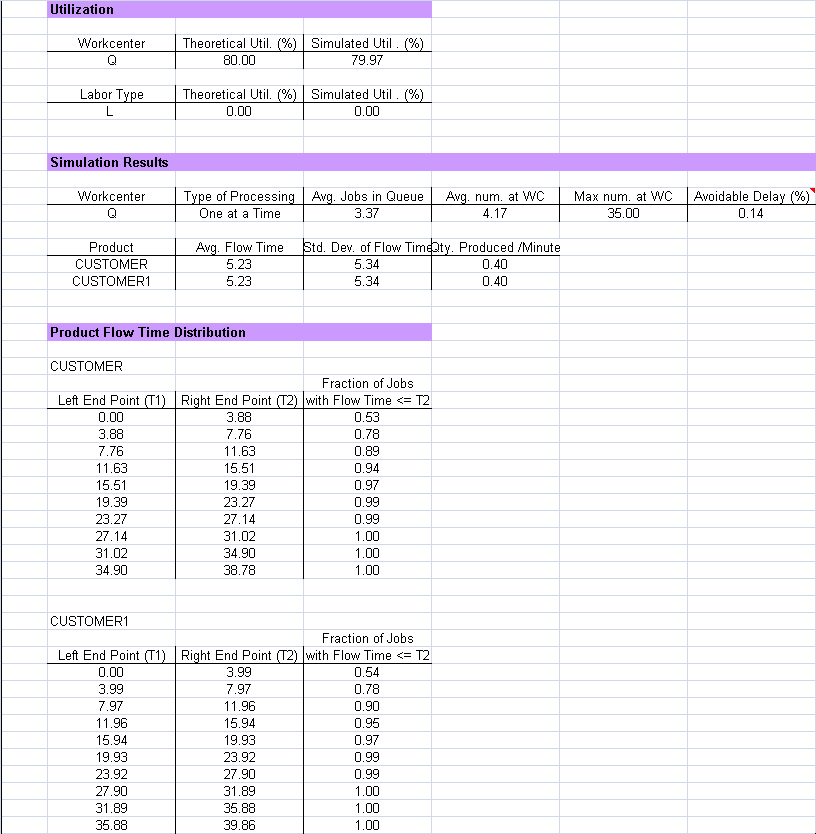
The simulation shows:



**14.3 The M/M/1 Queue with Priority**

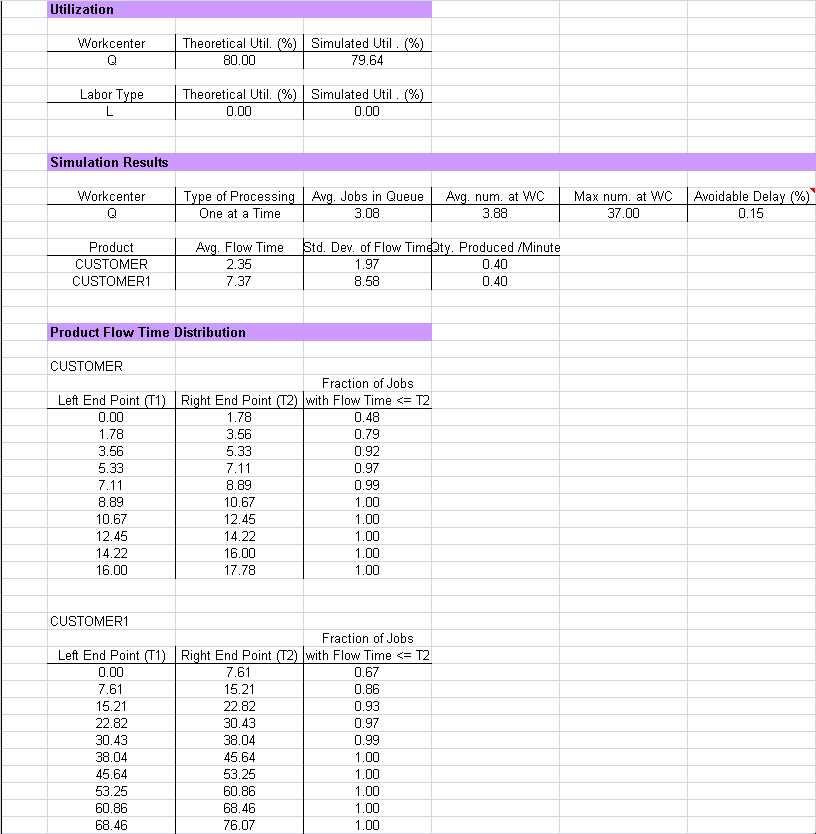
The file *mm1example.xls* contains the data for an M/M/1 queue with two types of customers. The arrival rate of each class of customers is 0.4 customer per minute. The average machine run time is 1 min., while the setup time is zero. There is zero labor run time or setup time. (We could as well have modeled this as zero run and setup time for the machine, and average run time for labor equal to 1 minute. It is important to observe that the run time cannot be positive for both machine and labor—please see section 3.5 for the consequences of making the run time greater than zero for both types of resources.) The system models an M/M/1 queue because the SCV of demand as well as the machine run time have been set equal to one. Therefore, HOM will use an exponential distribution for generating the interarrival time as well as the run time; see section 9.

If both classes are given the same priority, the system behaves like an M/M/1 queue with arrival rate = 0.4 + 0.4 = 0.8 customer per minute. The performance measures for the two customer classes are given below (simulation length = 200 days). The two classes are called CUSTOMER and CUSTOMER1.



Notice that the performance measures are almost identical for both customer classes and coincide with those in an M/M/1 queue that has an utilization of 0.8 (i.e., 80 percent). (Recall the average number in queue will be 0.8^2/(1-08) = 3.2. The average flow time will be number in system/arrival rate = (3.2 + 0.8)/0.8 = 5 min.

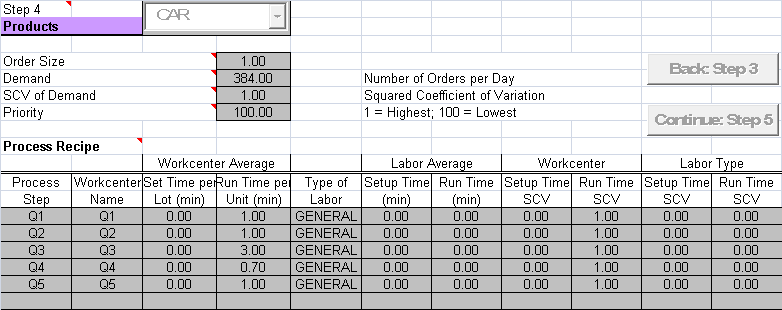
Modify the model such that the class CUSTOMER has priority of 1 (i.e., higher priority with respect to CUSTOMER1). (The user has to modify the model.) The output is shown below (simulation length = 200 days).



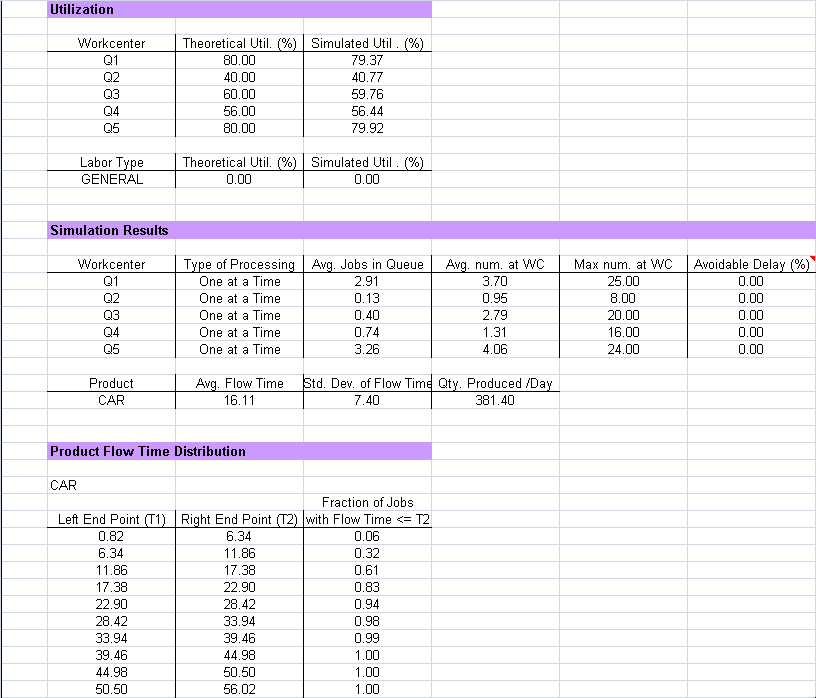
The class CUSTOMER now has smaller delay, whereas the class CUSTOMER1 has larger delay. The average delay of the two classes is the “same.”

**14.4 A Tandem Queue (or Assembly Line) Example**

The file *Tandem.xls* has the data for a five-stage queueing system. This could be a simplified model of an assembly system. The data is shown below, along with the theoretical utilization levels for the five stages. The interarrival and service time distributions are exponential. There are 1, 2, 4, 1, and 1 servers at the five stages. The recipe is given below. Can you guess the utilizations?



The output for this example is given below (for a simulation length of 200 days).



For this system, it turns out that each stage behaves like an M/M/C queue, with the number of servers C = 1, 2, 4, 1, and 1. The expected number of customers (average number at WC) at the five stages can be determined using the HOM: Waiting Line Management module, and are 4, 0.952, 2.831, 1.273, and 4, respectively. These numbers compare well with the simulated values!

**14.6 Circuit Board Manufacturing**

The file *circuit.xls* contains an example of circuit board manufacturing. There are three types of products, representing small, medium, and large orders. The process recipes are almost identical for the three products, except that the large product uses CNC machines. The priority for small orders is the highest (1). The priority has been set equal to 100 for the remaining types of orders. Demand as well as all processing times are deterministic (SCV = 0).

This model can be used to carry out complex what-if analyses, such as what happens if demand is no longer deterministic (try setting demand scv = 0.3 and examine effect on flow time), what is the impact of setting different product priorities (see the impact on flow time), what happens if the maximum lot size is reduced on some workcenters (how to do this? By reducing the maximum number of jobs that be processed at a time. Try reducing lot size in some workcenters and study impact on flow time), what is the “best” number of workers (can we reduce the number of workers? By how many?), can small orders be given high priority when large orders are quoted with a delivery leadtime of 1 month, and so forth.

**References**

Bohn, R.; K. Sommers and Greenberg *Kristen’s Cookie Company (A)* Boston, MA: President and Fellows of Harvard College, 1986.

*Donner Company*. Boston, MA: President and Fellows of Harvard College, 1988.

Law, Averill M., and W. David Kelton *Simulation Modeling & Analysis.* 2nd ed. New York, NY: McGraw-Hill, 1991.

Loch, C.; D. Grant and M. Harrison *Manzana Insurance—Fruitvale Branch*. Stanford, CA: Board of Trustees of the Leland Stanford Junior University, 1989.

Ross, S. M. *A Course in Simulation.* New York, NY: Macmillan, 1990.

Edited last: Sridhar Seshadri October 13 2017.