Mistakes and Misunderstandings:

DT Error

Prepared for the
MIT System Dynamics in Education Project
Under the Supervision of
Dr. Jay W. Forrester

by
Lucia Breierova
January 12, 1998

1. THE SYSTEM

Widgetmaster, Inc., decides how many widgets to produce in a certain period of time based on the number of orders for widgets the company expects to receive for that period. The managers of Widgetmaster compare the actual number of orders received to the number of orders they expected to receive, and then adjust their expectations for the future over a certain time period. Mathilda would like to know how quickly the managers of Widgetmaster adjust their expectations if the demand for widgets suddenly changes.

2. A FIRST ATTEMPT TO MODEL THE SYSTEM

Mathilda builds a simple model showing the relation between "ORDERS," the actual number of orders for widgets that Widgetmaster receives, and "Expected Orders," the number of orders for widgets that the managers expect to receive. The model is depicted in Figure 1.
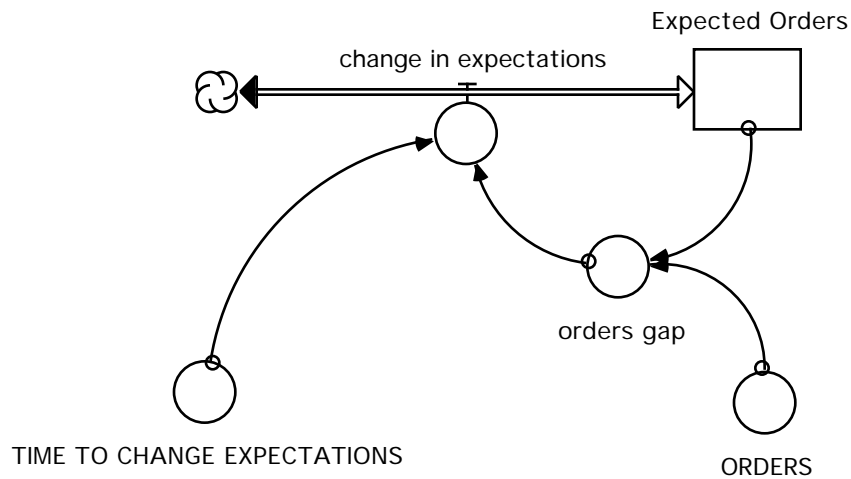

Figure 1: Model of "Expected Orders" for widgets

The stock "Expected Orders" changes through the flow "change in expectations." The managers' "change in expectations" equals the "orders gap," the difference between the actual "ORDERS" and the "Expected Orders," divided by the time constant "TIME TO CHANGE EXPECTATIONS."

Mathilda called the managers of Widgetmaster and found out that they usually receive orders for 200 widgets every month. "ORDERS" are therefore initially 200 widgets/month. To observe the dynamic behavior of the system, Mathilda sets the

"ORDERS" to suddenly step up to 300 widgets/month after the first month of the simulation. The managers also told Mathilda that they react quickly to a sudden change in the demand for widgets: they change their expectations in about three days. Mathilda therefore sets the "TIME TO CHANGE EXPECTATIONS" to three days, or 0.1 months.[1] Figure 2 shows the behavior of the system when Mathilda runs the model.

Figure 2: Behavior of the model

Mathilda is confused when she sees the behavior of "Expected Orders." If it only takes the managers of Widgetmaster three days to change their expectations, why do "Expected Orders" fluctuate so quickly between such high and low values over the course of four months?

3. MISTAKES AND MISUNDERSTANDINGS

The behavior of "Expected Orders" shown in curve 1 of Figure 2 is due to an error in the computation process during the simulation, known as DT error. In the real-world system, the expectations of Widgetmaster's managers would not show such fluctuations—the computer model produces them because of a computational error.

The solution interval, DT, is the time interval at which the simulation software solves the model equations and calculates the values of all variables. Any change in the model takes at least a time period equal to DT to occur. When Mathilda simulated the

---

[1] Please note that the "time to change expectations" equal to 0.1 months is unreasonably short for any real-life situation and is only used in this paper for illustrative purposes. A more reasonable value for the time constant would be a half month or a month.

model, the solution interval was set to 0.25 months, the default value in STELLA. The calculations are thus performed every quarter of a month, or four times each month. The managers of Widgetmaster, however, change their expectations every 0.1 months, or ten times every month. The value of 0.1 months for "TIME TO CHANGE EXPECTATIONS" is smaller than the solution interval, causing the inaccurate behavior. When the "ORDERS" change, the managers of Widgetmaster change their expectations 0.1 months later, but the the value of "Expected Orders" changes only after 0.25 months. The computational procedure therefore makes the flow of "change in expectations" too large, and hence the value of "Expected Orders" overshoots the value of "ORDERS." During the next solution interval, the computational procedure again reacts more slowly than Widgetmaster's managers to correct for the negative "orders gap." The "change in expectations" flow is then even larger than before (but has a negative value), so the value of "Expected Orders" falls below its initial value. The computational procedure again overcompensates too slowly, resulting in larger and larger fluctuations in the value of "Expected Orders," as shown in Figure 2,

Mathilda writes out the equations of the model and traces through the computation process:

Expected_Orders(t) = Expected_Orders(t - dt) + (change_in_expectations) * dt
INIT Expected_Orders = ORDERS
UNITS: widgets/month

INFLOWS:
change_in_expectations = orders_gap / TIME_TO_CHANGE_EXPECTATIONS
UNITS: (widgets/month)/month

ORDERS = 200 + STEP(100,1)
UNITS: widgets/month

orders_gap = ORDERS - Expected_Orders
UNITS: widgets/month

TIME_TO_CHANGE_EXPECTATIONS = 0.1
UNITS: month

Until time = 1 month, the system is in equilibrium because the value of "Expected Orders" equals the value of "ORDERS." Then, at time = 1 month, "ORDERS" step up to 300 widgets/month, but the "Expected Orders" have not changed yet. The "orders gap" is therefore equal to 100 widgets/month. Consequently, the flow of "change in expectations" is equal to 100/0.10, or 1000 (widgets/month)/month. The flow, however, has the value 1000 for only one DT, or 0.25 months. Thus, the change in the "Expected Orders" between time = 1 month and time = 1.25 months is 1000 * 0.25, or 250

widgets/month. Hence the value of "Expected Orders" jumps up to 200 + 250 = 450 widgets/month at time = 1.25 months.   Because the "ORDERS" are still 300 widgets/month, the "orders gap" is now –150 widgets/month, resulting in a flow of "change in expectations" equal to –1500 (widgets/month)/month between time = 1.25 months and time = 1.50 months. Hence, the stock "Expected Orders" changes by –1500 * 0.25, or –375 widgets/month, to fall to 75 widgets/month at time = 1.50 months. Continuing the calculations in a similar way, Mathilda obtains the table shown in Figure 3.

Months	Expected Orders	ORDERS	orders gap	change in expectations
.00	200.00	200.00	0.00	0.00
.25	200.00	200.00	0.00	0.00
.50	200.00	200.00	0.00	0.00
.75	200.00	200.00	0.00	0.00
1.00	200.00	300.00	100.00	1,000.00
1.25	450.00	300.00	-150.00	-1,500.00
1.50	75.00	300.00	225.00	2,250.00
1.75	637.50	300.00	-337.50	-3,375.00
2.00	-206.25	300.00	506.25	5,062.50
2.25	1,059.38	300.00	759.38	-7,593.75
2.50	-839.06	300.00	1,139.06	11,390.62
2.75	2,008.59	300.00	-1,708.59	-17,085.94
3.00	-2,262.89	300.00	2,562.89	25,628.91
3.25	4,144.34	300.00	-3,844.34	-38,443.36
3.50	-5,466.50	300.00	5,766.50	57,665.04
3.75	8,949.76	300.00	-8,649.76	-86,497.56
Final	-12,674.63	300.00	12,974.63	

Figure 3: Computation of "Expected Orders" with DT = 0.25 months and "TIME TO CHANGE EXPECTATIONS" = 0.1 months

The reader should trace out the computation to properly understand the computational procedure and its relationship to the model equations.

4. FIRST ATTEMPT TO CORRECT THE MISTAKE

To correct the DT error, Mathilda decides to reduce the solution interval to a half of its default value, or 0.125 months. Figure 4 shows the resulting behavior of the model.

Figure 4: Behavior of the model with DT = 0.125 months and "TIME TO CHANGE EXPECTATIONS" = 0.1 months

Even though the growing fluctuations of "Expected Orders" no longer occur, the behavior of the model is still unrealistic. The stock "Expected Orders" still repeatedly overshoots and then undershoots the desired value of 300 widgets/month, but now the fluctuations become smaller and smaller. The stock of "Expected Orders" eventually reaches the equilibrium value of 300 widgets/month. Again, Mathilda carries out the computation process, noting that now the computation is performed eight times each month, while Widgetmaster's managers still change their order expectations ten times each month. The computational procedure is still slower than the managers of Widgetmaster at correcting for an "orders gap." Because the solution interval is shorter than in the simulation that produced Figure 2, however, the value of the flow of "change in expectations" is not as large, so the overshoot and subsequent undershoot of "Expected Orders" are smaller. Hence, the fluctuations become smaller and smaller until the value of "Expected Orders" equals the value of "ORDERS."

Until time = 1 month, the system is in equilibrium. At time = 1 month, "ORDERS" step up to 300 widgets/month, while "Expected Orders" remain at 200 widgets/month, leading to an "orders gap" of 100 widgets/month. The flow "change in expectations" is therefore 1000 (widgets/month)/month, but only during one solution

interval, from time = 1 month to time = 1.125 months. Accordingly, the change in the "Expected Orders" stock is 1000 * 0.125 = 125 widgets/month between time = 1 month and time = 1.125 months, resulting in a value of 325 widgets/month at time = 1.125 months. Now, the "orders gap" is –25 widgets/month, and the flow "change in expectations" is –250 (widgets/month)/month. The change in "Expected Orders" is hence –250 * 0.125 = –31.25 widgets/month between time = 1.125 months and time = 1.250 months. The value of "Expected Orders" therefore falls to 293.75 widgets/month at time = 1.250 months. Continuing in a similar way, Mathilda calculates that the stock of "Expected Orders" reaches the equilibrium value of 300 widgets/month around time = 2 months, as summarized in the table in Figure 5 (only the calculations from time = 1 month to time = 2.5 months are shown; the system is in equilibrium at all other times).

Months	Expected Orders	ORDERS	orders gap	change in expectations
1.000	200.00	300.00	100.00	1,000.00
1.125	325.00	300.00	-25.00	-250.00
1.250	293.75	300.00	6.25	62.50
1.375	301.56	300.00	-1.56	-15.62
1.500	299.61	300.00	0.39	3.91
1.625	300.10	300.00	-0.10	-0.98
1.750	299.98	300.00	0.02	0.24
1.875	300.01	300.00	-0.01	-0.06
2.000	300.00	300.00	0.00	0.02
2.125	300.00	300.00	-0.00	-0.00
2.250	300.00	300.00	0.00	0.00
2.375	300.00	300.00	-0.00	-0.00
2.500	300.00	300.00	0.00	0.00

Figure 5: Computation of "Expected Orders" with DT = 0.125 months and "TIME TO CHANGE EXPECTATIONS" = 0.1 months

5. OVERCOMING OUR MISTAKES AND MISUNDERSTANDINGS

The computational error known as the DT error can be easily avoided by comparing the length of the solution interval, DT, to the shortest first-order delay in the system. In the system of Widgetmaster's order expectations, the shortest (and only) first-order delay is the "TIME TO CHANGE EXPECTATIONS" time constant. If the

solution interval is longer than the first-order delay, the simulation will produce inaccurate behavior, as in Figures 2 and 4. Therefore, the solution interval should always be shorter than the shortest first-order delay in the system. This section will try to determine how short the DT should be in comparison with the delay.[2]

If the solution interval is equal to the shortest first-order delay in the system, the simulation results in the behavior shown in Figure 6.

Figure 6: Behavior of the model with DT = 0.1 months and "TIME TO CHANGE EXPECTATIONS" = 0.1 months

The behavior shown in Figure 6 is a special case. The stock "Expected Orders" has jumped exactly to its desired value equal to "ORDERS" in just one solution interval, at the end of the solution interval following the step increase of "ORDERS." "ORDERS" increase to 300 widgets/month at time = 1.1 months, so the "orders gap" is equal to 100 widgets/month at time = 1.1 months. The "change in expectations" flow is therefore 1000 (widgets/month)/month from time = 1.1 months to time = 1.2 months, resulting in an increase of 1000 * 0.1 = 100 widgets/month in "Expected Orders," to a final value of 300 widgets/month. Thus, at time = 1.2 months, the "orders gap" becomes 0, and the system is at equilibrium. The behavior in Figure 6, however, is still not a realistic representation of the real-world system. The managers of Widgetmaster are unlikely to

---

[2] Another way to avoid the DT error is to change the Integration Method in the Time Specs dialog box in STELLA. While the default Euler's Method, which was used in the simulations shown in Figures 2 and 4, performs one calculation during each solution interval, the Runge-Kutta 2 and Runge-Kutta 4 methods perform, respectively, 2 and 4 calculations during each solution interval, thus effectively reducing the length of the solution interval. If the shortest first-order delay is extremely short, however, it is still

change their "Expected Orders" exactly to the new value of "ORDERS" immediately after "ORDERS" increase. Rather, they adjust their expectations more gradually, only after they observe a more permanent change in "ORDERS."

If the solution interval is shorter than the shortest first-order delay in the system (for example 0.0625 months,[3] with "TIME TO CHANGE EXPECTATIONS" = 0.1 months), the model produces the behavior shown in Figure 7.

Figure 7: Behavior of the model with DT = 0.0625 months and "TIME TO CHANGE EXPECTATIONS" = 0.1 months

The behavior shown in Figure 7 shows a gradual approach of "Expected Orders" to the equilibrium value of 300 widgets/month. The "ORDERS" step up to 300 widgets/month at time = 1 month, so the flow of "change in expectations" is equal to 1000 (widgets/month)/month between time = 1 month and time = 1.0625 months. The change in the value of "Expected Orders" during the first DT following the increase in "ORDERS" is therefore 1000 * 0.0625 = 62.5 widgets/month, resulting in the "Expected Orders" being equal to 262.5 widgets/month at time = 1.0625 months. During the next time interval, the "orders gap" is reduced further. The model thus does not cause an overshoot and subsequent undershoot of the goal value of the "Expected Orders," and the stock asymptotically approaches 300 widgets/month.

---

possible that the solution interval will not be reduced enough by simply changing the integration method.

When the solution interval is halved one more time, to 0.03125 months, the model produces the behavior shown in Figure 8.

Figure 8: Behavior of the model with DT = 0.03125 months and "TIME TO CHANGE EXPECTATIONS" = 0.1 months

The behavior in Figure 8 closely resembles that in Figure 7, but the approach of "Expected Orders" to equilibrium is more gradual when the solution interval is shorter, as in Figure 8. Because the solution interval is equal to one half of the solution interval used to produce the simulation in Figure 7, twice as many calculations must be performed.

It is important to realize that a model is always only a representation of the real world. In a real-world system, the behavior is continuous over time, whereas a simulation model only performs the calculations at distinct time points. A continuous behavior can be approximated by using an extremely short solution interval; the trade-off, however, is a longer time required to simulate the model.

---

[3] Because of the binary arithmetic of computers, it is recommended to set the solution interval to a number that comes from the sequence $(1/2^n)$ (i.e., a number that can be obtained by repeatedly dividing 1 by 2). The numbers from this sequence are 1, 0.5 (or 1/2), 0.25 (or 1/4), 0.125 (or 1/8), 0.0625 (or 1/16), 0.03125 (or 1/32), ...

6. KEY LESSONS

Before simulating a model, the modeler should make sure that a time constant with a small value will not introduce a DT error. If DT error does occur, the modeler should look for the shortest first-order delay in the system and set the solution interval accordingly. When simulating a model, the modeler should always try to repeatedly cut the DT in half to check whether the behavior generated by the model is caused by the model structure rather than by an error associated with the solution interval. In any model, the solution interval DT should be less than a half of the shortest first-order delay in the system.

7. APPENDIX: MODEL DOCUMENTATION

☐ **Expected_Orders(t)** = Expected_Orders(t - dt) + (change_in_expectations) * dt
        INIT Expected_Orders = ORDERS
        DOCUMENT: Number of orders Widgetmaster expects to receive each month.
        Units: widgets/month

        ☞ INFLOWS:
        change_in_expectations = orders_gap / TIME_TO_CHANGE_EXPECTATIONS
        DOCUMENT: Rate of change in Widgetmaster's "Expected Orders."
        Units: (widgets/month)/month

○ **ORDERS** = 200 + STEP(100,1)
        DOCUMENT: Orders for widgets produced by Widgetmaster. After the first
        month, the orders suddenly step up by 100 widgets per month.
        Units: widgets/month

○ **orders_gap** = ORDERS - Expected_Orders
        DOCUMENT: The difference between the actual number of widgets ordered and
        the number of widgets that Widgetmaster's managers expect to be ordered.
        Units: widgets/month

○ **TIME_TO_CHANGE_EXPECTATIONS** = 0.1
        DOCUMENT: Time in which Widgetmaster's managers change their
        expectations about the orders Widgetmaster receives.
        Units: month

8. BIBLIOGRAPHY

Jay W. Forrester, 1968. Principles of Systems. Portland, OR: Productivity Press.

Steve Peterson and Barry Richmond, 1993. STELLA II: Technical Documentation.
        High Performance Systems, Inc.