CHAPTER 4

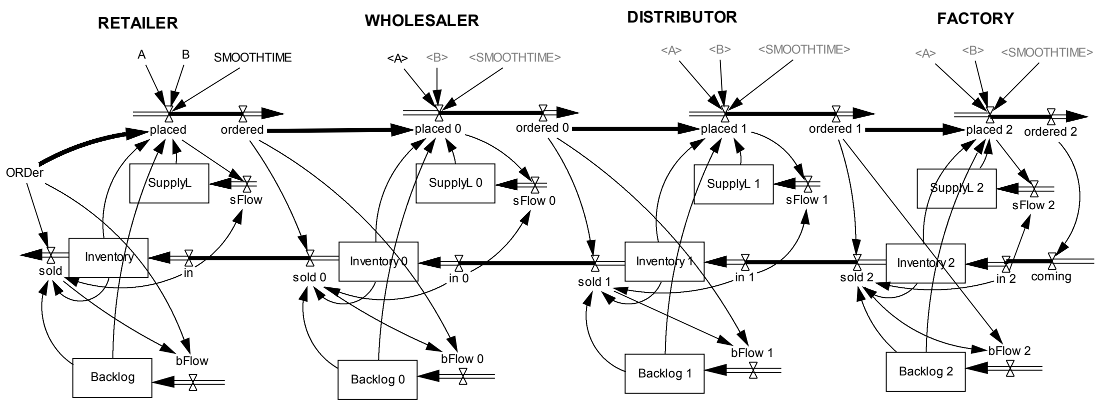
The Beer Game

This chapter presents the stock and flow diagram and Vensim equations for a system dynamics model of the Beer Game. Senge (1990) and Sterman (1989) discuss this game further. The model presented here is based on the approach outlined in Sterman (1989). The primary purpose of this chapter is to document the model for discussion and analysis. Because of this, there is only limited analysis of the model and the rationale behind it.

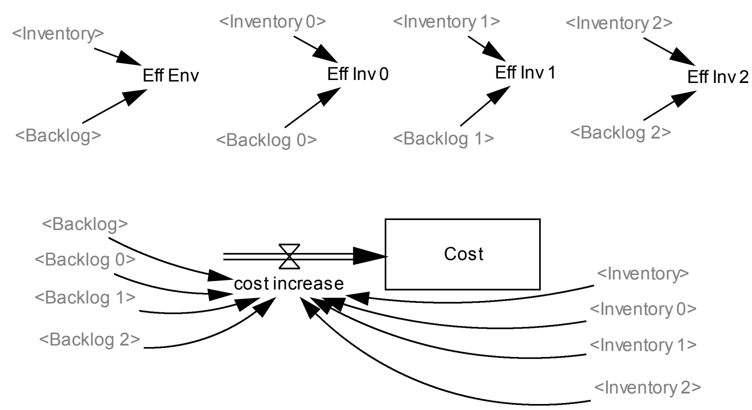
4.1 Stock and Flow Diagram

The primary portion of the stock and flow diagram is presented in Figure 4.1a. Scorekeeping aspects of the model which collect information about the model for output purposes are presented in Figure 4.1b. Note that this model was created using the replicate feature of the Vensim system. Specifically, the Retailer sector of the model was created, and then this was replicated to create the Wholesaler, Distributor, and Factory sectors. When replication is done using Vensim, the names in the original section are automatically changed by adding a number to them. Thus, the names in the Wholesaler sector are the same as those in the Retailer sector except that they have a 0 (zero) appended to them. Similarly, the names in the Distributor and Factory sectors have a 1 and a 2, respectively, added to them.

After the sectors were replicated, they were connected together as shown in Figure 4.1a. The heavier information arrows and solid material flow pipes represent system elements which have delays. The constants A and B are defined in the system equations presented in the next section, and these are based on the modeling approach presented in Sterman (1989). SMOOTHTIME is the smoothing time for a first order exponential smoothing operation used to represent the averaging process used in the ordering process for each sector.



a. Part 1



b. Part 2

Figure 4.1 *Beer game stock and flow diagram*

4.2 Vensim Equations

The equations are mostly straightforward bookkeeping for the flows of materials and orders. The ordering policies are modeled in equations 33 through 36. Equations 47 through 50 keep track of the orders placed, but not yet received (the supply line), for each sector.

```
(01) A = 0.25
(02) B = 0.33
(04) Backlog 0 = INTEG(bFlow 0,0)
(05) Backlog 1 = INTEG(bFlow 1,0)
(06) Backlog 2 = INTEG(bFlow 2,0)
(07) bFlow = ORDer - sold
```

```
(08) bFlow 0 = ordered - sold 0
(09) bFlow 1 = ordered 0 - sold 1
(10) bFlow 2 = ordered 1 - sold 2
(11) coming = ordered 2
(12) Cost = INTEG(cost increase, 0)
(13) cost increase
        = 1 * (Backlog + Backlog 0 + Backlog 1 + Backlog 2)
          + 0.5 * (Inventory + Inventory 0
                   + Inventory 1 + Inventory 2)
(14) Eff Env = Inventory - Backlog
(15) Eff Inv 0 = Inventory 0 - Backlog 0
(16) Eff Inv 1 = Inventory 1 - Backlog 1
(17) Eff Inv 2 = Inventory 2 - Backlog 2
(18) FINAL TIME  = 36
(19) in = DELAY FIXED(sold 0, 2, 4)
(20) in 0 = DELAY FIXED(sold 1, 2, 4)
(21) in 1 = DELAY FIXED(sold 2, 2, 4)
(22) in 2 = DELAY FIXED(coming, 2, 4)
(23) INITIAL TIME  = 0
(24) Inventory = INTEG(in - sold, 12)
(25) Inventory 0 = INTEG(in 0 - sold 0, 12)
(26) Inventory 1 = INTEG(in 1 - sold 1, 12)
(27) Inventory 2 = INTEG(in 2 - sold 2, 12)
(28) ORDer = 4 + STEP(4, 5)
(29) ordered = DELAY FIXED(placed, 1, 4)
(30) ordered 0 = DELAY FIXED(placed 0, 1, 4)
(31) ordered 1 = DELAY FIXED(placed 1, 1, 4)
(32) ordered 2 = DELAY FIXED(placed 2, 1, 4)
(33) placed = MAX(0,
        SMOOTH(ORDer, SMOOTHTIME)
          + A * (12 - (Inventory - Backlog) - B * SupplyL))
(34) placed 0 = MAX(0,
        SMOOTH(ordered, SMOOTHTIME)
          + A * (12 - (Inventory 0 - Backlog 0) - B * SupplyL 0))
(35) placed 1 = MAX(0,
        SMOOTH(ordered 0, SMOOTHTIME)
          + A * (12 - (Inventory 1 - Backlog 1) - B * SupplyL 1))
(36) placed 2 = MAX(0,
        SMOOTH(ordered 1, SMOOTHTIME)
          + A * (12 - (Inventory 2 - Backlog 2) - B * SupplyL 2))
(37) SAVEPER  = TIME STEP
(38) sFlow = placed - in
(39) sFlow 0 = placed 0 - in 0
(40) sFlow 1 = placed 1 - in 1
(41) sFlow 2 = placed 2 - in 2
(42) SMOOTHTIME = 1
(43) sold = MIN(Inventory + in, ORDer + Backlog)
```

```
(44) sold 0 = MIN(Inventory 0 + in 0, ordered + Backlog 0)
(45) sold 1 = MIN(Inventory 1 + in 1, ordered 0 + Backlog 1)
(46) sold 2 = MIN(Inventory 2 + in 2, ordered 1 + Backlog 2)
(47) SupplyL = INTEG(sFlow, 0)
(48) SupplyL 0 = INTEG(sFlow 0, 0)
(49) SupplyL 1 = INTEG(sFlow 1, 0)
(50) SupplyL 2 = INTEG(sFlow 2, 0)
(51) TIME STEP  = 1
```

4.3 Plots

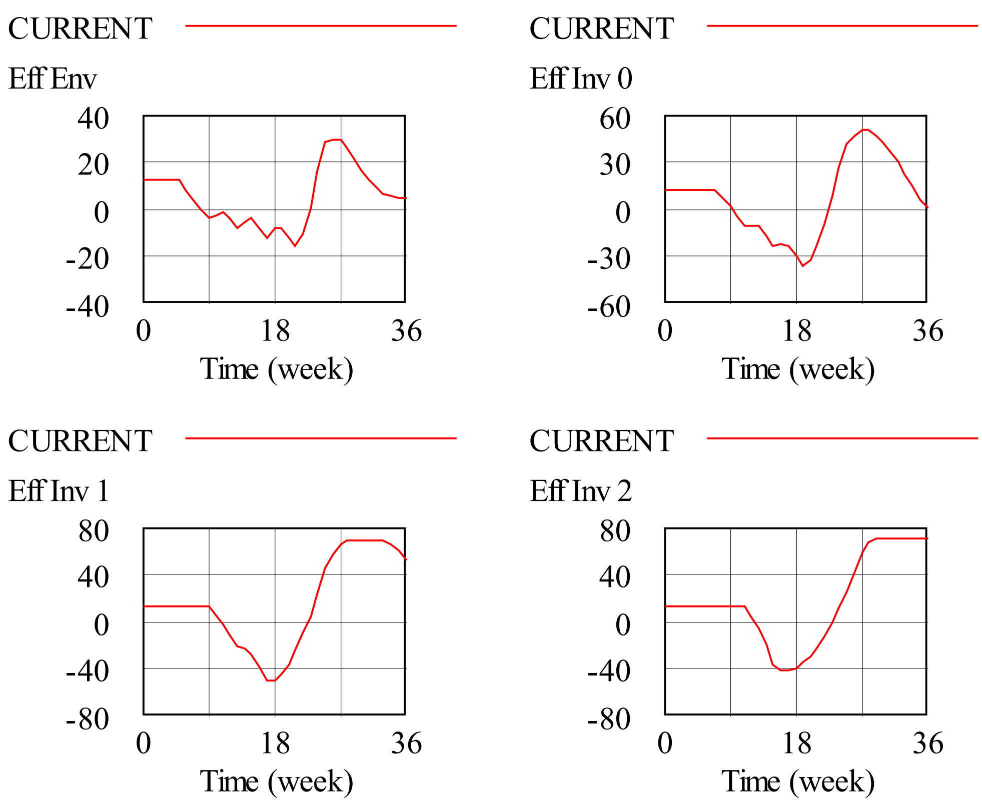
Figure 4.3a shows plots of the effective inventory for each sector of the model. This shows a qualitative pattern similar to that observed in actual plays of the game.

Figure 4.3b shows a plot of the cumulative total cost for all sectors. The final cost of $2,250 is close to the average seen in actual plays of the beer game. Sterman (1989) suggests that the model for the Factory sector should be somewhat modified from that for the other sectors, and this has not been done in the model presented above.

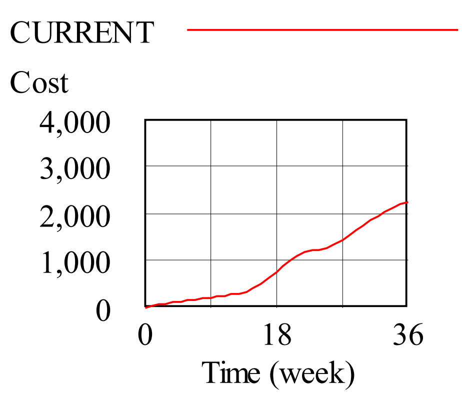
4.4 References

P. M. Senge, *The Fifth Discipline: The Art and Practice of the Learning Organization*, Doubleday Currency, New York, 1990.

J. D. Sterman, Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment, *Management Science*, Vol. 35, pp. 321 329 (1989).



a. Effective inventory



b. Cost

Figure 4.3 *Plots for model run*