Equation support for MISQ Article

Thomas D. Clark, Jr.
Louisiana State University
E. J. Ourso College of Business Administration
Baton Rouge, LA 70803-5256
Telephone: 225 578-9076
Fax: 225 578-2511
tomclark@lsu.edu

Mary C. Jones
University of North Texas
College of Business Administration
Denton, TX 76203
Telephone: 940.565.3167
Fax: 940.565.4935
jonesm@unt.edu

1. **Abstract:** This document provides an illustration of a System Dynamics approach to assessment of a theoretical model of management support systems (MSS). It illustrates the way in which system dynamics can be used to assess models other than those of physical systems for which it normally is used. It is an example of an indexed System Dynamics (SD) model of the common structures of managerial support systems. The structure of the model is shown in Figure 1 in the schematic format for dynamic equations using the iThink approach.[1] It is the translation of the causal diagram of the system's structure shown in Figure 2. As can be seen in Figure 1, all of the reservoir or level variables are indexed with bi-flow (both directions) rates. This indexed approach is employed to explore a purely theoretical structure, although this structure can be documented and at least partially verified from previous studies of parts of it.[2] This approach to theory development is somewhat at variance with the normal system dynamics approach that deals almost exclusively with physical systems. The true value of the model is its ability to expose a theory about the system's structure using an approach that provides a vehicle for experimentation with various hypotheses about its behavior. The purpose of such experimentation is to enhance insight into the behavior of the system that stems from its structure and suggest management polices that might be introduced into it, thus improving our understanding of how to design and manage such systems.

Place Figures 1 and 2 About Here

There are many in the SD community who believe that all system dynamics models must solve a specific problem. This is a more narrow view of the science than taken here as the example model was not structured with any given problem in mind but with the view that a theoretical structure not linked to a specific problem has great value, especially in its ability to improve insight into system behavior. Using this approach, system dynamics becomes a valuable tool in exposing the underlying theoretical

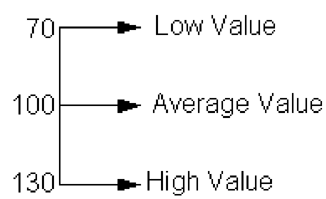
[1] iThink is the modeling approach used for this paper. Several other SD approaches are available but all have the same inherent equation structure based on reservoirs or level variables and the rates of flow into and out of them. All use the first order difference equation format. For a complete discussion of the methodology see Stearman 2000.

[2] See Clark, et al 2007, MISQ paper.

structure of abstract systems. The primary purpose of this paper is to explore the methodology behind this and to discuss the approach taken. In doing so, the characteristics of the system involved also will be advanced and discussed.

2. Reservoir (Level) Equations

The following set of equations is for the reservoirs or levels of the system. The key features of management support systems, regardless of specific type, are captured in these variables. These reservoir variables, shown in Figure 1 with the rectangle symbol ( ), include the quality of the MSS, how well it matches the problem space in which it operates, the ease with which i[illegible]e used, how rapidly and well it responds to data requests and problem solution processes, its inherent functionality, management's and users' commitment to its development, and the knowledge base of the people using the systems. Also shown are the elements that influence these variables which include the costs to develop and operate the system, the amount of training the system requires and the technology available to develop the system. These inherent or generic characteristics also are shown in the causal model of Figure 2. The complex, information feedback structure of the system can be seen in both of the diagrams. It is this complex feedback structure that makes such systems difficult to manage and evaluate, and why the ability to experiment with various policy options using a dynamic model gives the model both theoretical and practical value. Each is shown with its initial value and the rates of flow into and out of the accumulators. The initial value of each of the reservoirs is set at 100, the neutral or average value. The index structure for each of them is:



To illustrate the manner in which each is structured and to show their value, a simple illustration will be used to facilitate the discussion. It is shown in figure 3 where the reservoir is the indexed variable, the rate of change is bidirectional and the inputs to the rate represent positive and negative changes in its value.

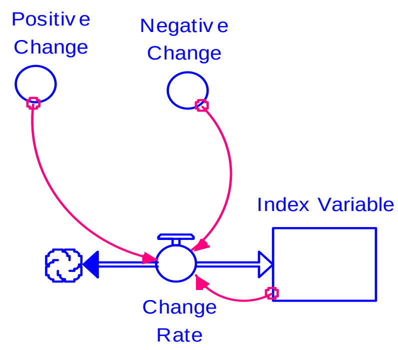


Figure 3. Indexed Variable Structure.

The equations for this structure are:

```
Index Variable (t) = Index Variable (t - dt) + (Change_Rate) * dt
INIT Index Variable = 100
Change_Rate = Index_Variable*(Negative_Change+Positive_Change)
Negative_Change = if time < 52 then 0.0 else if time > 52 and time < 115
then (-0.0005) else 0.0
```

The input for positive change is a sine wave that revolves around 0.0 and returns values form 0.0005 to -0.0005. The negative effects are illustrated using a function that is 0.0 until time period 52 (one year) when it becomes -0.0005 until time period 115 when it returns to zero. The behavior resulting can be seen in figure 4.

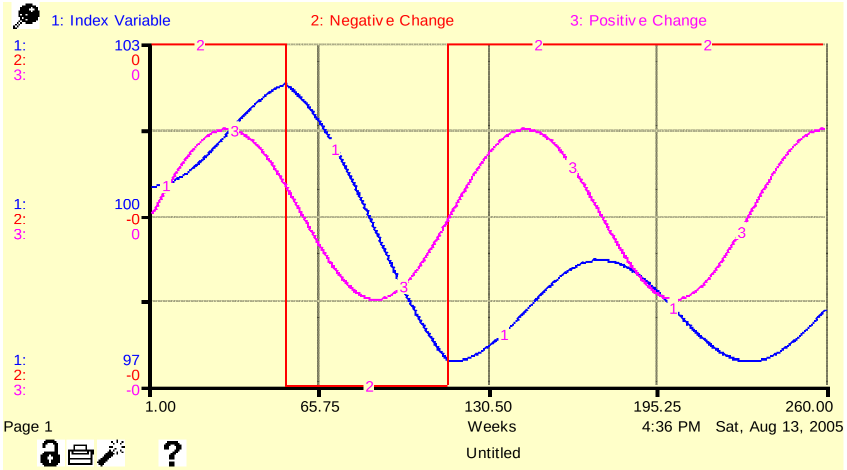


Figure 4. Output Under Experimental Conditions I

The indexed variable follows the sine wave input until period 52 when it begins to rapidly decline because of the larger negative influence. It stabilizes at a value between a value between 97 and 98 with the sine wave structure with about the same lag as before. If the negative influence is removed, the resulting behavior can be seen in figure 5 where the indexed variable follows the sine wave input with a lag of about six weeks. In an actual model, the input variables would be changed to represent

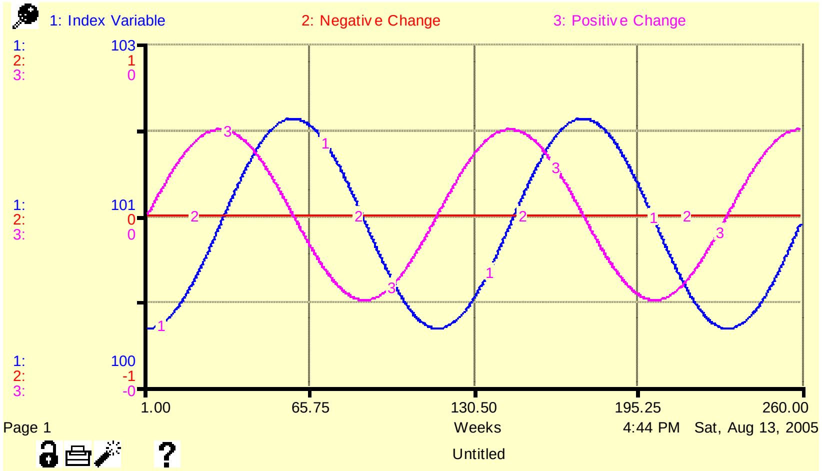


Figure 5. Output Under Experimental Conditions II

management policies or market conditions and the resulting behaviors analyzed. The behaviors under varying conditions provide an expanded understanding of the theories about system behavior. While this brief introduction illustrates the structure, it is more instructive to look at the actual model and its behavior. To accomplish this we will begin by discussing the reservoir variables in the model and their determinants. After this discussion, several experiments with the model will be introduced with discussion about the theoretical implications. The other variables in the model that transfer information to and between the levels and rates are presented in a later discussion.

2.1 Executive Commitment to the MSS.

Executive_Committment_to_the_MSS(t) = Executive_Committment_to_the_MSS(t - dt) + (EC_Change_Rate) * dt

INIT Executive_Committment_to_the_MSS = 100
EC_Change_Rate = Executive_Committment_to_the_MSS*(Benefits_Factor+Cost_Factor+Gap_Factor)

2.2 Level of MSS Use.

Level_of_MSS_Use(t) = Level_of_MSS_Use(t - dt) + (MSS_Use_Change_Rate) * dt
INIT Level_of_MSS_Use = 100
MSS_Use_Change_Rate = Level_of_MSS_Use*(Functionality_Factor+PSC_Factor+UC_Factor)

2.3 Management Decision Quality.

Management_Decision_Quality(t) = Management_Decision_Quality(t - dt) + (EDQ_Change_Rate) * dt
INIT Management_Decision_Quality = 100
EDQ_Change_Rate = Management_Decision_Quality*(Level_of_Use_Factor+PSC_Factor+Quality_Factor)

2.4 MSS Costs.

MSS_Costs(t) = MSS_Costs(t - dt) + (Costs_Change_Rate) * dt
INIT MSS_Costs = 100
Costs_Change_Rate = MSS_Costs*(MSS_Use_Factor+Tech_Base_Factor+Training_Factor+UID_Factor)

2.5 MSS Desired.

MSS_Desired(t) = MSS_Desired(t - dt) + (MSS_Desired_Change_Rate) * dt
INIT MSS_Desired = 100
MSS_Desired_Change_Rate = MSS_Desired*(EC_Factor+UC_Factor)

2.6 MSS Functionality and MSS Technology Base. 2004).

MSS__Tech_Base(t) = MSS__Tech_Base(t - dt) + (TB_Change_Rate) * dt
INIT MSS__Tech_Base = 100
TB_Change_Rate = (if((Tech_Base_Needed > 0.0) and (Tech_Base_Needed < Development_Potential)) then Tech_Base_Needed else Development_Potential)+(TB_Decay_Factor*MSS__Tech_Base)

MSS_Functionality(t) = MSS_Functionality(t - dt) + (Functionality_Change_Rate) * dt
INIT MSS_Functionality = 100
Functionality_Change_Rate = MSS_Functionality*Tech_Base_Factor

2.7 MSS_Knowledge_Base_Required & User's_Knowledge Base.

MSS_Knowledge_Base_Required(t) = MSS_Knowledge_Base_Required(t - dt) + (KBR_Change_Rate) * dt
INIT MSS_Knowledge_Base_Required = 100

KBR_Change_Rate = MSS_Knowledge_Base_Required*(KB_Factor+PSC_Factor+Tech_Base_Factor)

User's_KB(t) = User's_KB(t - dt) + (UKB_Change_Rate) * dt
INIT User's_KB = 100
UKB_Change_Rate = User's_KB*(MSS_Use_Factor+Training_Factor+UID_Factor)

2.8 MSS Problem Space Match.

MSS_Problem_Space_Match(t) = MSS_Problem_Space_Match(t - dt) + (MSS_to_PSM_Change_Rate) * dt
INIT MSS_Problem_Space_Match = 100
MSS_to_PSM_Change_Rate = MSS_Problem_Space_Match*(Functionality_Factor+PSC_Factor+UID_Factor)

2.9 MSS Quality.

MSS_Quality(t) = MSS_Quality(t - dt) + (MSS_Quality_Change_Rate) * dt
INIT MSS_Quality = 100
MSS_Quality_Change_Rate = MSS_Quality*(Functionality_Factor+MSS_PSC_Match+Usability_Factor)

2.10 MSS Training.

MSS_Training(t) = MSS_Training(t - dt) + (Training_Change_Rate) * dt
INIT MSS_Training = 100
Training_Change_Rate = MSS_Training*KBR_Factor

2.11 Perceived MSS Benefits.

Perceived_MSS_Benefits(t) = Perceived_MSS_Benefits(t - dt) + (Benefits_Change_Rate) * dt
INIT Perceived_MSS_Benefits = 100
Benefits_Change_Rate = (MDQ_Factor+PSC_Factor)*Perceived_MSS_Benefits

2.12 Usability of the MSS.

Usability_of_the_MSS(t) = Usability_of_the_MSS(t - dt) + (Usability_Change_Rate) * dt
INIT Usability_of_the_MSS = 100
Usability_Change_Rate = Usability_of_the_MSS*(KB_Factor+Tech_Base_Factor)

2.13 User Commitment to the MSS.

User_Committment_to_the_MSS(t) = User_Committment_to_the_MSS(t - dt) + (UC_Change_Rate) * dt
INIT User_Committment_to_the_MSS = 100

UC_Change_Rate = User_Committment_to_the_MSS*(Benefits_Factor+Cost_Factor+KB_Factor+Usability_Factor)

2.14 User Involvement in Development.

User_Involvement_in_Development(t) = User_Involvement_in_Developemnt(t - dt) + (UID_Change_Rate) * dt
INIT User_Involvement_in_Developemnt = 100

UID_Change_Rate = User_Involvement_in_Developemnt*(EC_Factor+UC_Factor)

3. Connector Variables

The following set of equations provides inputs for the system, represents decision points in the system, transfers information about variables form one point to another or links one variable to another. They are called connectors in the iThink framework. In some cases, extended discussion of a variable is not necessary because the equation is self explanatory. The following two connectors are discussed to provide clarification on the logic of how connectors are built. Again a more detailed explanation of the logic underlying the connectors can be found in Clark, et al. 2005.

3.1 Market Technology Available. The Market Technology Available variable is structured to provide for periodic increases in the technology available in the marketplace for development or enhancement of the system. It provides for two different approaches to input the technology available. In the form shown here, the equation has a step increase of 10% at just over two years into the simulation period. A variety of forms that mimic technology growth are possible.

Market_Technology_Available = ((If(time<=52) then 100 else if (time>52 and time <108) then 110 else if (time >108 and time < 260) then 120 else 100)*0.0)+(100+step(15,78)+step(10,108)+step(10,175))

3.2 Problem Space Complexity. As discussed earlier, problem space complexity is a critical variable in the model as it contributed to the perceived benefits of the system, management’s decision quality, the knowledge required by managers to operate in the problem space and the extent to which the MSS might be used. In the current form it is input as a sine wave with an amplitude of 5 and a period of 115 weeks. This form is used to reflect the trajectory from more to less complex space depending on market conditions. A number of input forms are possible.

Problem_Space_Compexity = SINWAVE(5,115)

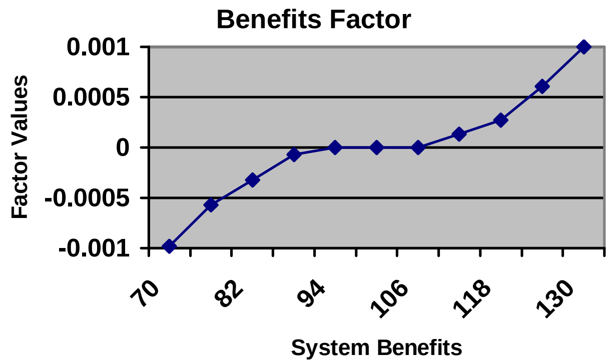
TB_Decay_Factor = -0.001

Technology_Gap = Market_Technology_Available-MSS__Tech_Base

4. Table Functions

The following set of equations translates the values of the reservoir variables into factors that are used to calculate other variables. Each is a function of the reservoir variable it translates. All of them have essentially the same structure so only one will be discussed in detail to illustrate how they are formed. As can be seen in the graph here, a value of 70 in the Perceived MSS Benefits causes the factor to be strongly negative and a value of 130 would cause it to be strongly positive. The factor is neutral in the values around 94 to 106. Then it slowly rises or falls depending upon the value of the independent variable, which in this case is Perceived MSS Benefits.

Benefits_Factor = GRAPH(Perceived_MSS_Benefits)
(70.0, -0.00098), (76.0, -0.00057), (82.0, -0.00032), (88.0, -7e-005), (94.0, 0.00), (100, 0.00), (106, 0.00), (112, 0.00013), (118, 0.00027), (124, 0.00061), (130, 0.001)



Cost_Factor = GRAPH(MSS_Costs)

(70.0, 0.00098), (76.0, 0.00057), (82.0, 0.00023), (88.0, 0.00014), (94.0, 4e-005), (100, 0.00), (106, 0.00), (112, 0.00), (118, -0.000205), (124, -0.000415), (130, -0.00099)

Development_Potential = GRAPH(Market_Technology_Available-MSS_Desired)
(-20.0, 5e-006), (-13.0, 0.00035), (-6.00, 0.000425), (1.00, 0.00075), (8.00, 0.001), (15.0, 0.00123), (22.0, 0.0017), (29.0, 0.00255), (36.0, 0.0039), (43.0, 0.00435), (50.0, 0.005)

EC_Factor = GRAPH(Executive_Committment_to_the_MSS)
(70.0, -0.00099), (76.0, -0.0004), (82.0, -0.00022), (88.0, -9e-005), (94.0, -1e-005), (100, 0.00), (106, 0.00), (112, 0.0002), (118, 0.00035), (124, 0.0005), (130, 0.001)

Functionality_Factor = GRAPH(MSS_Functionality)
(70.0, -0.001), (76.0, -0.00059), (82.0, -0.00029), (88.0, -0.00014), (94.0, 0.00), (100, 0.00), (106, 6e-005), (112, 0.00015), (118, 0.00031), (124, 0.00059), (130, 0.001)

Gap_Factor = GRAPH(Technology_Gap)
(-50.0, -0.00099), (-40.0, -0.00051), (-30.0, -0.00027), (-20.0, -0.0001), (-10.0, -4e-005), (0.00, 0.00), (10.0, 0.0001), (20.0, 0.00018), (30.0, 0.00037), (40.0, 0.00062), (50.0, 0.001)

KBR_Factor = GRAPH(MSS_Knowledge_Base_Required)
(70.0, -0.001), (76.0, -0.00055), (82.0, -0.00027), (88.0, -0.00013), (94.0, 0.00), (100, 0.00), (106, 5e-005), (112, 0.00012), (118, 0.00022), (124, 0.00052), (130, 0.001)

KB_Factor = GRAPH(User's_KB)
(70.0, -0.00099), (76.0, -0.00063), (82.0, -0.0004), (88.0, -0.00018), (94.0, 0.00), (100, 0.00), (106, 6e-005), (112, 0.00017), (118, 0.00026), (124, 0.00049), (130, 0.00099)

Level_of_Use_Factor = GRAPH(Level_of_MSS_Use)
(70.0, -0.00096), (76.0, -0.0006), (82.0, -0.00035), (88.0, -0.00012), (94.0, -2e-005), (100, 0.00), (106, 6e-005), (112, 0.00027), (118, 0.00046), (124, 0.0006), (130, 0.00097)

MDQ_Factor = GRAPH(Management_Decision_Quality)
(70.0, -0.001), (76.0, -0.00059), (82.0, -0.00032), (88.0, -0.00012), (94.0, 0.00), (100, 0.00), (106, 5e-005), (112, 0.00015), (118, 0.00038), (124, 0.00064), (130, 0.00099)

MSS_PSC_Match = GRAPH(MSS_Problem_Space_Match)
(70.0, -0.00099), (76.0, -0.00058), (82.0, -0.00031), (88.0, -0.00011), (94.0, 0.00), (100, 0.00), (106, 0.00), (112, 0.00018), (118, 0.00042), (124, 0.00074), (130, 0.001)

MSS_Use_Factor = GRAPH(Level_of_MSS_Use)
(70.0, -0.00099), (76.0, -0.00057), (82.0, -0.00023), (88.0, -0.0001), (94.0, 0.00), (100, 0.00), (106, 0.00), (112, 0.00015), (118, 0.00035), (124, 0.0007), (130, 0.00097)

PSC_Factor = GRAPH(Problem_Space_Compexity)
(-10.0, -0.001), (-8.00, -0.0006), (-6.00, -0.00021), (-4.00, -6e-005), (-2.00, 0.00), (0.00, 0.00), (2.00, 4e-005), (4.00, 0.00013), (6.00, 0.00034), (8.00, 0.00059), (10.0, 0.001)

Quality_Factor = GRAPH(MSS_Quality)
(70.0, -0.001), (76.0, -0.00061), (82.0, -0.00039), (88.0, -0.00017), (94.0, 0.00), (100, 0.00), (106, 3e-005), (112, 0.00013), (118, 0.0004), (124, 0.00066), (130, 0.001)

Tech_Base_Factor = GRAPH(MSS__Tech_Base)
(70.0, -0.001), (76.0, -0.0006), (82.0, -0.00039), (88.0, -0.0002), (94.0, 0.00), (100, 0.00), (106, 6e-005), (112, 0.00013), (118, 0.00027), (124, 0.0006), (130, 0.001)

Tech_Base_Needed = GRAPH(MSS_Desired-MSS__Tech_Base)
(-00.0, -0.00097), (-13.0, -0.00028), (-6.00, -1e-005), (1.00, 0.00017), (8.00, 0.00041), (15.0, 0.00086), (22.0, 0.00143), (29.0, 0.00221), (36.0, 0.00317), (43.0, 0.00428), (50.0, 0.00497)

Training_Factor = GRAPH(MSS_Training)
(70.0, -0.00097), (76.0, -0.00055), (82.0, -0.00033), (88.0, -0.00013), (94.0, 0.00), (100, 0.00), (106, 7e-005), (112, 0.00019), (118, 0.00042), (124, 0.00072), (130, 0.00099)

UC_Factor = GRAPH(User_Committment_to_the_MSS)
(70.0, -0.00099), (76.0, -0.00052), (82.0, -0.00022), (88.0, -6e-005), (94.0, 0.00), (100, 0.00), (106, 0.00013), (112, 0.00023), (118, 0.00033), (124, 0.0005), (130, 0.00099)

UID_Factor = GRAPH(User_Involvement_in_Developemnt)
(70.0, -0.00099), (76.0, -0.00062), (82.0, -0.00038), (88.0, -0.00017), (94.0, -6e-005), (100, 0.00), (106, 7e-005), (112, 0.00015), (118, 0.00037), (124, 0.00063), (130, 0.00099)

Usability_Factor = GRAPH(Usability_of_the_MSS)
(70.0, -0.001), (76.0, -0.0006), (82.0, -0.00037), (88.0, -0.00011), (94.0, 0.00), (100, 0.00), (106, 6e-005), (112, 0.00014), (118, 0.00032), (124, 0.00061), (130, 0.00098)

The foregoing discussion illustrates a methodology for developing a theoretical system dynamics model using prior research as a basis for development. The resulting model then provides a theoretical structure for the referent system. Such systems are very difficult if not impossible to study as a whole because of their very complex information feedback nature. The model, however, can be operated to investigate various hypotheses about behavior and policies can be tested to evaluate various approaches to their management. The next section contains a discussion that illustrates the value of this approach.

5. Model Behavior

The first discussion is for behavior illustrating problem space complexity and the behavior in other variables that stems from it. To illustrate this behavior, an external sine wave function was used to establish the values of the Problem Space Complexity variable. This simulates a dynamic problem space that at times is very complex and at other times less so. A typical space might revolve around competitive markets, financial structure, or operations and logistics. As can be seen from the structure of the model and the previous discussion of complexity, this is a key variable that drives the behavior of the Perceived MSS Benefits, Management Decision Quality and the Level of MSS Use. The form of problem space complexity and the PSC factor are shown in Figure 6 and the behavior of the variables it drives in Figure 7. As seen in the figures, the variables that are dependent on the PSC Factor fluctuate in concert with the sine wave input of Problem Space Complexity. As the problem space becomes more complex, the MSS is perceived to offer greater benefits, the decision maker uses in more, and managers make higher quality decisions.

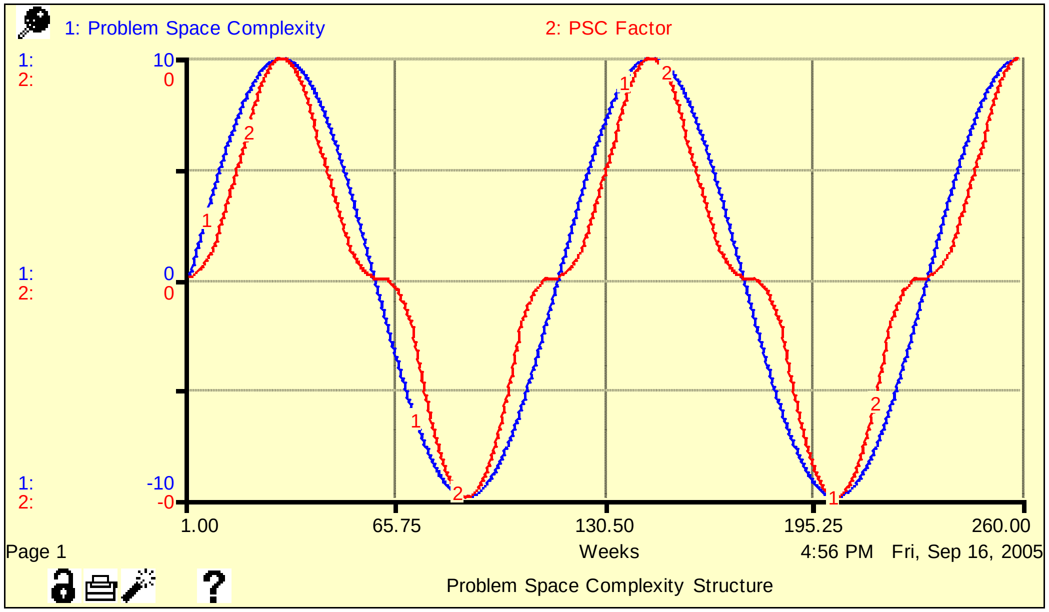


Figure 6. Problem Space Complexity Behavior.

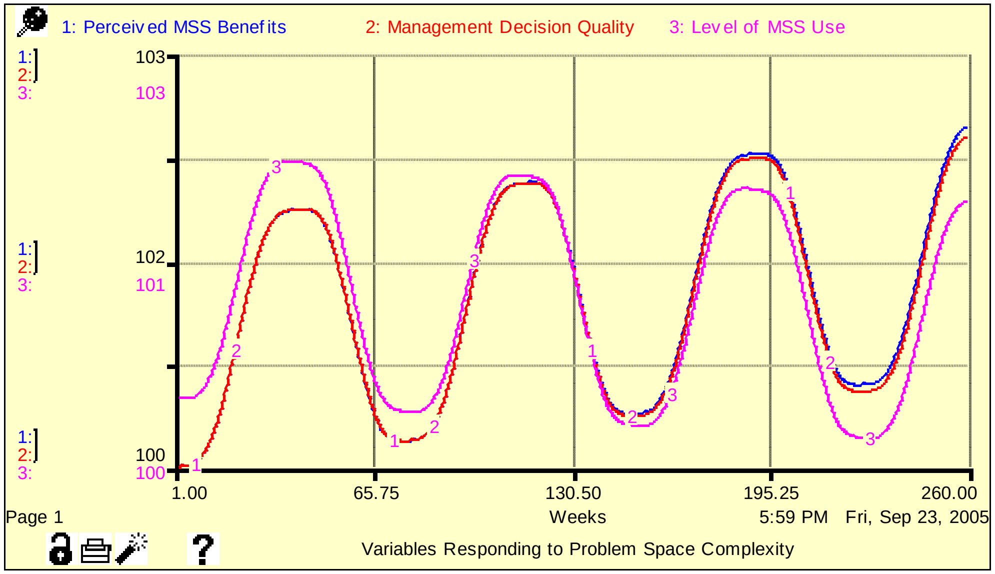


Figure 7. PSC Response Variables Behavior

The effect that these variables translate into the remainder of the model can be seen by observing the behavior of the variable Executive Commitment to the MSS, which also is affected by the Technology Gap, Perceived Benefits and MSS Costs (see Figure 1). Its behavior is shown in Figure 8. Executive Commitment, in this case, responds primarily to the increasing gap between the technology in use by the organization and what is available in the market place to its competitors. The other factors are stable so have no effect. If, however, technology costs increase, the resulting behavior can be seen in Figure 9. A 15% increase in costs causes the Executive Commitment to fall until the 156th week when the Technology Gap finally overcomes the cost increase. The declining Executive Commitment continues until the gap reaches a point where investment becomes necessary which, in turn, causes the commitment variable to stabilize at a low point. This suggests that executive commitment to an MSS is exhibited when the MSS requires more of the executives' attention or activity to retain or enhance its usefulness (for example, when new technology enters the market). However, a sharp increase in costs decreases commitment to the MSS; as perhaps the executives abandon it or look for an alternative. This is a key area because it suggests that at some point in the life of an MSS, executives will no long support it, or at least will no longer support breathing new life into it. Understanding where this point is can be useful in establishing policy about managing the MSS.

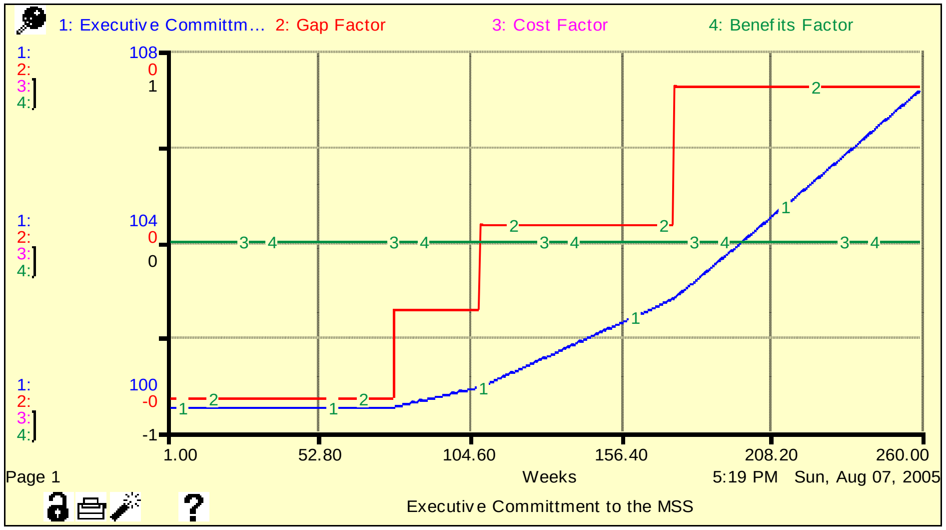


Figure 8. Executive Commitment to the MSS Behavior

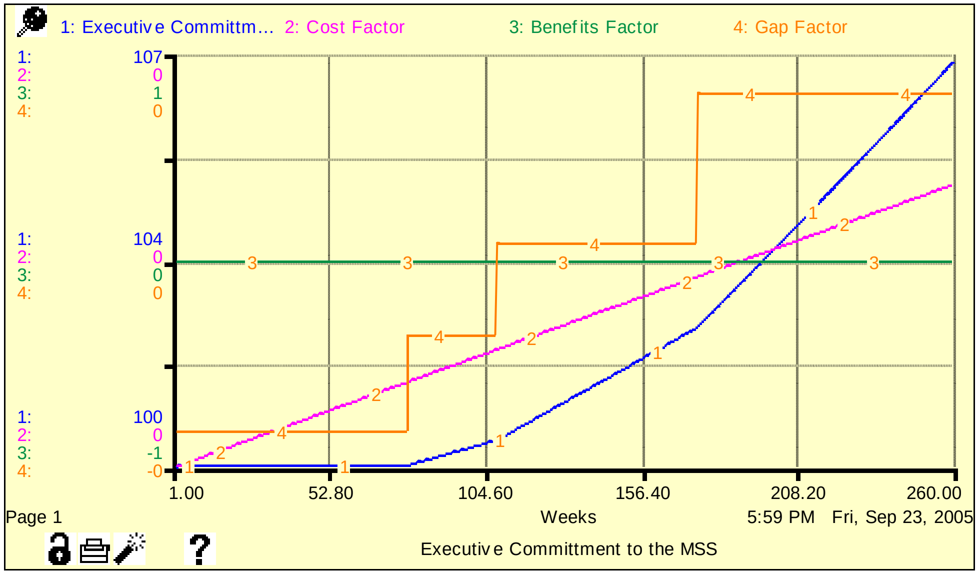


Figure 9. Response to a 15% Increase in Costs

The behavior of Management Decision Quality (Variable 1) can be seen in the graph of Figure 10. Management Decision Quality follows the sine wave driven input of Problem Space Complexity (Variable 3) and the Level of Use (Variable 4). As Problem Space Complexity rises, System Use begins to respond, followed by a corresponding response in Decision Quality. From the time that complexity peaks until the time that quality peaks, there is a lag of about nine or ten months. This out of phase behavior illustrates the difficulty in precisely matching a system to a problem space at a point in time and reinforces the need to design flexibility into the system (Sprague and Carlson, 1982). Flexibility here is included in the MSS Functionality and Usability constructs.

The rising level of decision quality is stimulated only slightly by the costs of the system falling, which is seen as variable 2 in Figure 10. This stems primarily from the relatively small drop in costs. Executives would be much more likely to use and develop systems that had high relative cost advantages. Also near the end of the simulation, the Level of Use moves into phase with Management Decision Quality. This too is a response to the falling costs and the increasing investment that follows. The investment slightly increases quality because the system becomes more functional as a result of the investment.

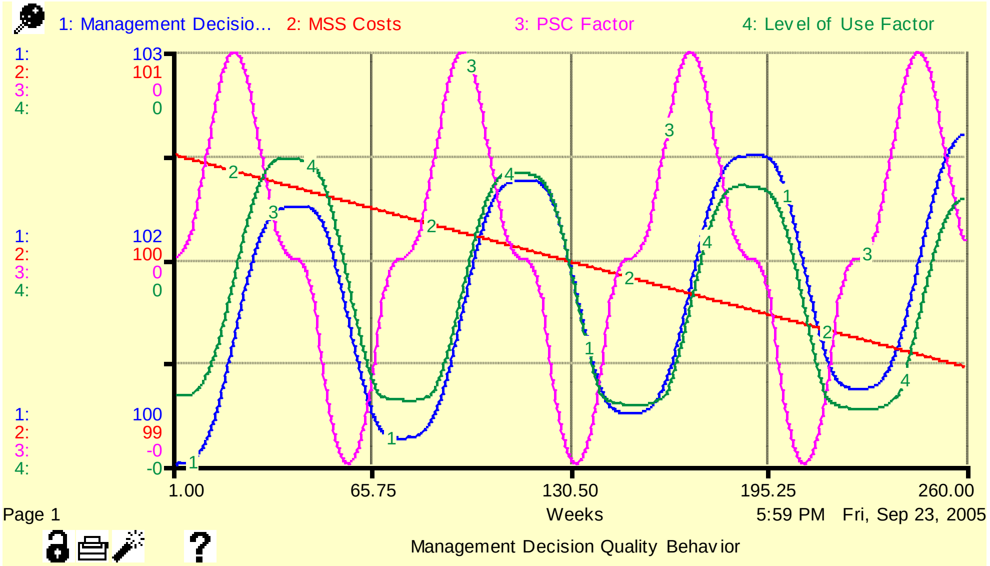


Figure 10. Management Decision Quality Behavior

Both of these discussions illustrate an approach that can be used to fully investigate system behavior. To illustrate how theory can be extended with the approach, a simple research question is proposed and addressed. Several times in the discussion of the model, Problem Space Complexity and System Functionality were discussed. An interesting research question about their interaction could be posed as: what is the relationship between MSS functionality and the match between the MSS and problem space? The behavior of the two variables under the initial conditions discussed earlier is shown in Figure 11 where variable 1 is MSS to Problem Space Match, variable 2 is MSS Functionality, variable 3 is User Involvement in Development, variable 4 is Problem Space and variable 5 is MSS Quality (see Figure 1 for the relationships posited). The sharp increase in user involvement occurs when both "functionality" and "problem space match" cross a rapid growth threshold. This suggests that as additional functionality is needed to close the gap between MSS and problem space, users respond. There is also an increase in "MSS quality" when there is a small regular investment in functionality. Thus, keeping the MSS 'current' improves its quality. The MSS to Problem Space Match (variable 1) tracks problem space complexity with a very slight increasing slope. The lag is about the same as previously discussed.

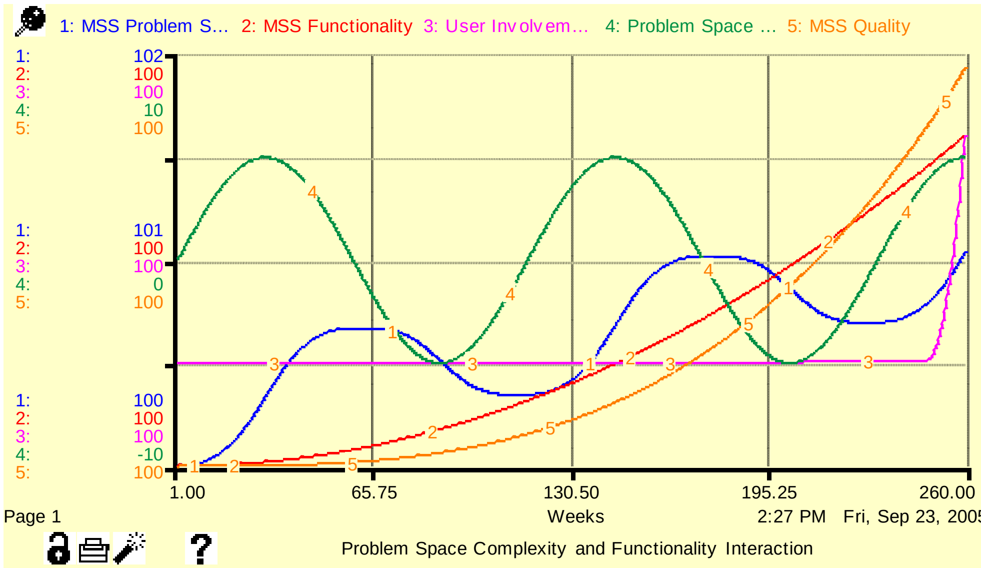
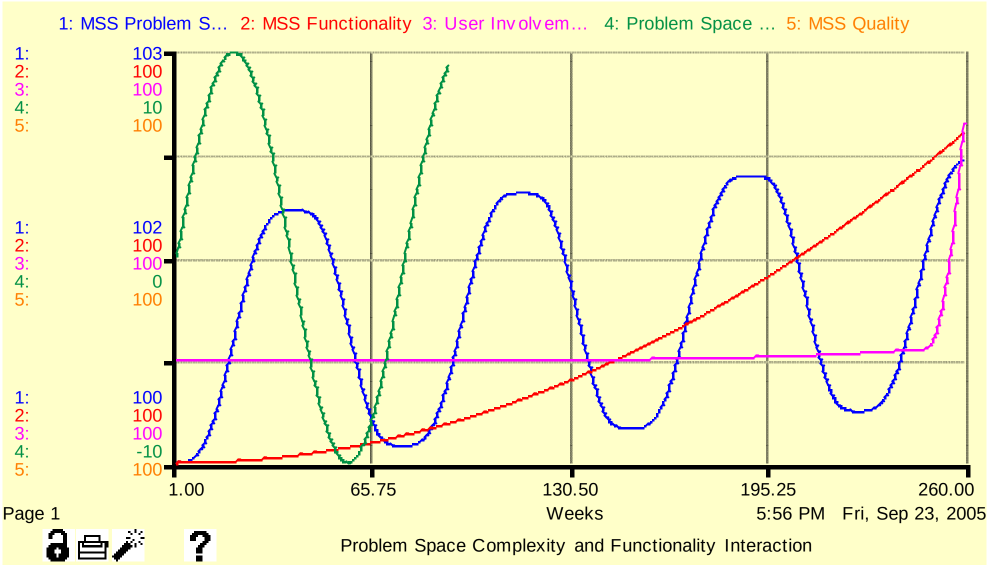


Figure 11. Problem Space Complexity and Functionality Interaction

In Figure 12, the behavior illustrated occurs when the problem space is made more variable by increasing its inherent complexity and shortening its cycle from 115 weeks to 75 weeks. The MSS to Problem Space Match exhibits the same basic pattern of behavior as before but responds more slowly to the changes. Quality and functionality display essentially the same behavior as shown in Figure 11, thus showing very little response to the problem space complexity change. As long as the functionality responds to the change, both quality and system to problem space match perform as always. However, if functionality is <u>not</u> enhanced to correspond to changes in the problem space complexity, it rapidly declines, impacting MSS quality dramatically (Figure 13). Interestingly, there is little direct effect on MSS to Problem Space Match although it is lower and the response lag is larger contributing to the quality decline. The MSS also responds less quickly to the change in problem space complexity. Therefore, while the change in problem space complexity variability has some effect on the match, it is minor compared to the more dramatic effects of a functionality decline and dramatic decline in MSS quality. This indicates that MSS management policy should include regular assessments of functionality and the problem space to determine where enhancements in the MSS are most needed.


1: MSS Problem S...
2: MSS Functionality
3: User Involvem...
4: Problem Space ...
5: MSS Quality
1: 103
2: 100
3: 100
4: 10
5: 100
1: 102
2: 100
3: 100
4: 0
5: 100
1: 100
2: 100
3: 100
4: -10
5: 100
1.00
65.75
130.50
195.25
260.00
Page 1
Weeks
5:56 PM Fri, Sep 23, 2005
Problem Space Complexity and Functionality Interaction