3 Bounded rational managerial role play and the generic systems it creates

The behavior of organizations and economies is driven by bounded-rational managerial role play. Thus, theoretical concepts based on rational agency must be replaced by models subsuming multiple down to earth managerial roles, whose performance calls for making use of limited information for taking actions that meet the short-term goals of the subsystems the role players are charged to manage. Organizational science attributes the formation and discharge of these roles to a variety of factors, such as rules, expectations of others in the organization as well as in the organizational environment, while they are also affected by external rewards, internalized motivation, and aspirations and personality of the role player (1).

Figure 3.1 illustrates the information processing and decision-making roles of the policy agents operating in a system of role-sending institutions. The decision rules used by the policy agents are formed by norms, values, expectations, and sometimes explicit rules emanating over the long term from the role-sending institutions. The decision process is based on access of the agents to information and the manifest as well as informal rules guiding their decisions. Clearly, this process constitutes a bounded rational rather than an absolute rational framework as has been pointed out in the seminal work of Herbert Simon (2).

An operational model of an economic system constructed as a test bed for policy design may include both the role-sending functions of the institutions involved in the process and the role-playing functions of the decision agents. The model boundary will of course depend on the problem of interest. When the causes of an institutional change are to be investigated, the long-term process of changes in rules and norms must be included in the model. However, when the short-term behavior of an organization is to be explored, the long-term processes can be excluded, since the behavior to be addressed by the model would arise from the role-play of its managers working under stable rules and norms (3).

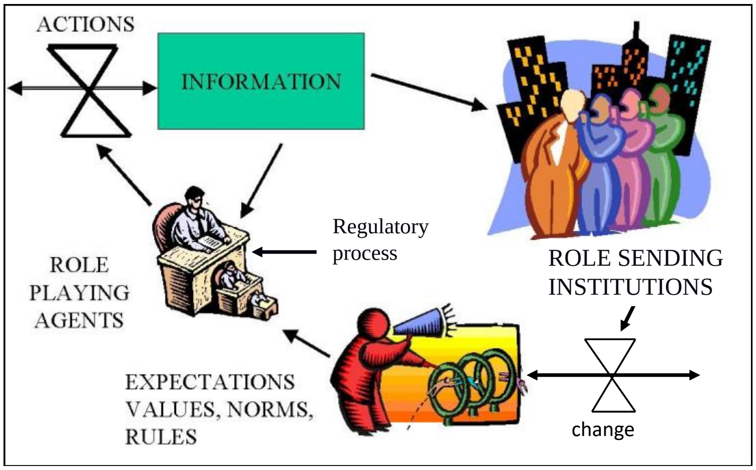


Figure 3.1 Managerial role-play as described in organizational sciences. (Source: (3))

These are albeit abstract concepts and not hugely useful in helping the task of modeling the bounded rational role system in which managers function. Such roles call for making iterative decisions to address the discrepancies from goals found in everyday operations (4), (2). Forrester took a very pragmatic view of managerial roles. He focused on realistically representing the actual decision-making practice in organizations for dealing with specific policy issues in his many modeling projects. This chapter breaks down the decision rules or policies he used into a small number of generic processes that can be combined to represent a wide variety of managerial decisions. It also discusses the basic systems formed by managerial role play that are pervasive across multiple problem domains.

The anatomy of managerial decisions

Managerial decisions might appear quite complex, especially as presented in the management curricula and myriads of books and papers on the various aspects of management. Although institutions like firms, markets, and economies consist of multiple managers making the best of limited information to discharge their respective roles without even knowing the exact consequences of their decisions, the mainstream economic theory views institutions as rational agents making optimal decisions based on perfect information and knowledge of the outcomes of those decisions. It should be recognized that managerial policies are, first of all, based on available information about past behavior and almost no knowledge of future. Second, they combine available information for creating decisions that meet their short term albeit changing goals. The conscious decision process involves observing the discrepancy between actual and intended conditions, then acting to overcome that discrepancy. Thus, goal, observation, discrepancy, and action are elements of managerial decisions (5).

The actions resulting from managerial decisions create flows that update the stocks of information driving those decisions. Taken in a bounded rational framework, managerial decisions are based on best judgement and are repeated following the same rules (4), which means that the iterative corrections they create can potentially lead to optimal outcomes over time, but this depends on the system structure that may create unintended patterns such as undesirable growth, unintended limits to desirable growth, and instability. Iterative corrections can also lead to dysfunctional homeostasis like low productivity, poor quality and long delivery delays (6); economic stagnation (7), (8); and income inequality (9), authoritarianism (10) and anarchy (11).

Managerial policies can also create reinforcing feedback processes when past performance drives expectations of the future motivating a repeat of the allocation rules in use. Examples include infrastructure expansion decisions in firms and multiplier mechanisms in economies. Reinforcing feedback may also result from a cascade of multiple bounded rational actions. For example, while population growth seems to be tied to the birth fraction, the birth fraction is an outcome of multiple individual actions seeking goals of family size, quality of life considerations, adherence to social norms, old age security, etc.

The bounded rational policies are not based on random logic but are a combination of a small number of decision rules. These policies may also not be denoted as logic statements as if driving a switching process in a digital computer but must be represented as continuous mathematical relationships subsuming decision patterns. Called processes, these relationships are discussed below:

1. Production

The production process calls for production of a good or service with a resource external to the conservative system to which the production flow is connected. As shown in Figure. 2, the production process can drive policies governing both an inflow into and outflow out of a stock. It is computed by multiplying the resource with its productivity.

Examples of production process include:

Weekly Production of a good or service = workers*goods or service produced per worker per week

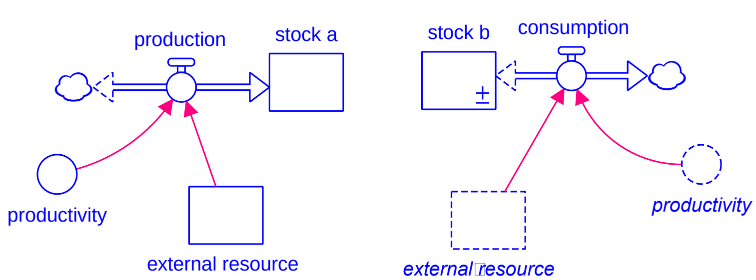
Yearly Food consumption = Population*food consumed per person per year

Monthly Wage payments = Employees*average salary per employee per month

Weekly Sale orders = Salesmen*orders booked per salesman per week

Monthly Interest payments = Debt*monthly interest rate

Monthly Crime rate = criminals*crimes committed per criminal per month



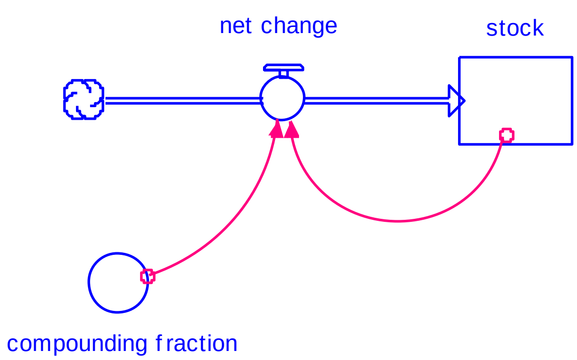
*production = external resource*productivity*
*consumption = external resource*productivity*

Figure 3.2 Decision rule mimicking production

When used to represent an outflow that must not drive a stock to a negative value, the production logic must often be combined with what is called first order control that must gradually inhibit drain when the stock falls below an acceptable value and stop it when it is empty. Examples include shipments out of an inventory, emigration out of a population, pumping out of a fluid reservoir, etc. The logic of combining production and first order control will be discussed under complex policies.

2. Compounding

This decision process results in compounding of the stock connected to the flow since the flow is determined by that stock, as shown in Figure 3.3. In its simplest form, the flow is calculated by multiplying the stock with a compounding fraction.



*Net change = stock*compounding fraction*

Figure 3.3 **Decision process creating compounding.**

Examples of compounding process include:

Births = Population*birth fraction

Increase in enthusiasm = Enthusiasm*enthusiasm growth fraction

Interest income = Bank balance*interest rate

In migration = Population*in migration fraction

3. Draining

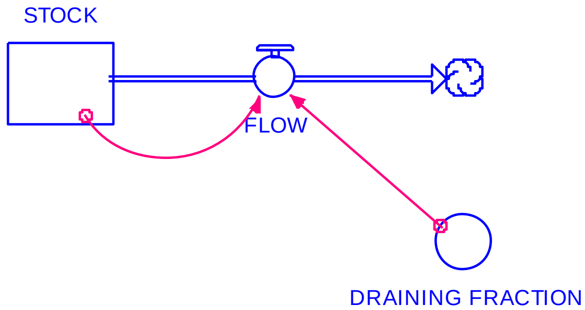
The logic of the draining process is like the compounding process, but as shown in Figure 3.4, the flow it creates drains the stock it is connected to. Its calculation entails multiplying the stock driving it by a draining fraction. Examples of draining include:

Deaths = Population/average life expectancy

Depreciation = Assets/asset life

Capital retirement = Capital*retirement fraction

Workforce attrition = Employees*attrition fraction



*FLOW = STOCK*DRAINING FRACTION*

Figure 3.4 Decision process creating draining.

Note the calculation of compounding as well as draining involves either multiplication of the resource by a fractional change over time or dividing it by a time constant. For example,

fractional death rate = 1/life expectancy

fractional asset depreciation = 1/asset life

fractional worker attrition rate = 1/average length of employment

4. Stock adjustment to maintain goals

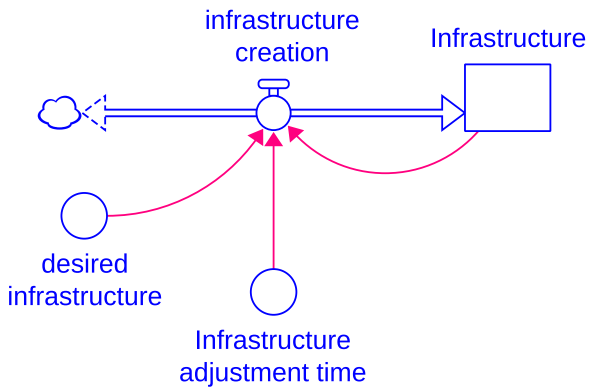
The fourth generic policy entails the use of an observed or perceived discrepancy between the managed stock and a goal condition that the management desires to maintain. This policy creates an adjustment that can be both positive or negative, depending on whether the stock exceeds the goal or falls short of it. Hence its rate of change will often be a bi-flow unless operational considerations call for constraining it in a specific way.

This policy is the mainstay of most managerial decisions, whether taken in a formal or informal role. Formal roles include maintaining an inventory, a workforce, a stock of machines, or any other infrastructure. Informal roles include maintaining relationships, refrigerator contents, grocery expenses, etc.

When goals can be explicitly discerned and, the discrepancy between a goal and existing condition of the stock can be computed as the arithmetic difference between the two, the policy takes the form of what is termed as 'Explicit Stock Adjustment' process. When the discrepancy can only be perceived indirectly through a symptom, adjustment becomes a matter of judgment, the policy takes the form of what is termed as 'implicit stock adjustment'. However, since the adjustment process is dynamic, precision in measurement of a discrepancy or its translation into an adjustment may not lead to serious problems. An over-adjustment or under-adjustment gets compensated in the subsequent round of decisions. Perception of the discrepancy in an indirect way however introduces delays that can create unforeseen dynamics. The logic of the explicit and implicit stock adjustment processes is discussed below:

Explicit stock adjustment process

The logic of explicit stock adjustment calls for precisely knowing the discrepancy between the goal condition and the current state and adjusting this discrepancy over a specific period of time as shown in Figure 3.5. Thus, when the goal or the desired level of a managed infrastructure can be explicitly measured, the adjustment creating the flow into the stock of the infrastructure is computed by dividing the discrepancy by a time constant. Division by a time constant is necessary since the discrepancy cannot be adjusted instantaneously and the time constant – whether long or short subsumes the time taken to defray the discrepancy.



Infrastructure creation = (desired infrastructure-Infrastructure)/Infrastructure adjustment time

Figure 3.5 Decision process incorporating adjustment towards an explicit goal.

Following examples illustrate the logic of this policy. Examples of explicit stock adjustment include adjustment of workforce, production infrastructure, inventories of goods, budgets, etc. Following examples illustrate the logic of this policy:

Workforce changes = (desired workers-workers)/workforce adjustment time

Capacity changes = (Desired capacity-capacity) * capacity adjustment fraction

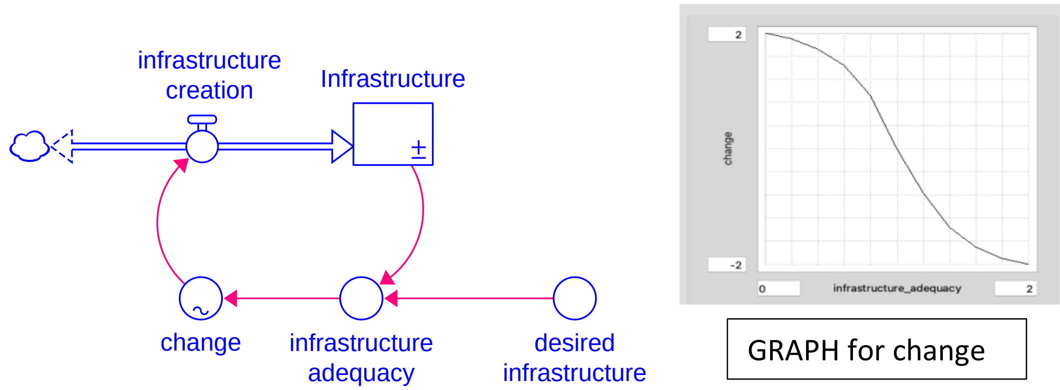
Widget orders = (Desired inventory- inventory)/inventory adjustment time

Change in grocery budget = (desired budget-current budget)/budget adjustment time

As in case of compounding and draining, explicit stock adjustment involves either multiplication of the discrepancy by a fractional change over some period of time or dividing it by a time constant. These parameters are either a function of the prevalent managerial practice or represent the aggressiveness of the managerial behavior. The adjustment time or adjustment fraction is thus a behavioral parameter determined by the role player. An aggressive response to discrepancy would return a short adjustment time and a laid-back response a long adjustment time.

Implicit Stock adjustment process

A stock adjustment policy is often not cognizant of an explicit goal but might be driven by a perception of adequacy of the infrastructure residing in a performance measure, like loss of customers, rise in complaints, idle capacity, perceptions of quality, delivery delay, processing time, etc. In such cases, the adjustment of flow is approximate, although the iterative process in which it occurs creates a movement towards goal that is similar to the explicit stock adjustment process. Also, since precise assessment of adjustment is difficult, it often manifests in adding or discarding a fraction of the existing infrastructure as illustrated in Figure 3.6. The implicit adjustment can subsume asymmetric adjustment for positive and negative discrepancies which is often the case.



infrastructure creation = change
change = GRAPH (infrastructure adequacy)
infrastructure adequacy = Infrastructure/desired infrastructure

Figure 3.6 Decision process incorporating adjustment towards an implicit goal.

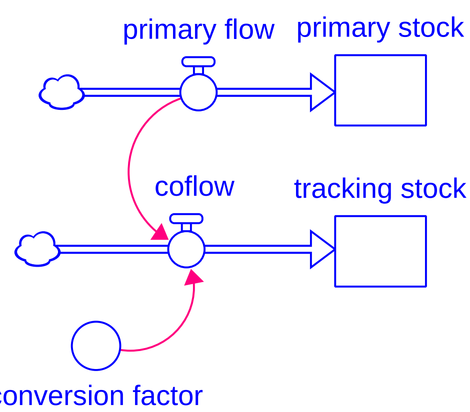
GRAPH can be represented as a graphical function built into most software. This function may use infrastructure adequacy as input and graphically represent its relationship to the change it invokes.

The stock adjustment process, called proportional control in engineering, lies at the core of the feedback control systems. It however suffers from problems like steady state error, and sluggish adjustment when the discrepancy is small. Multiple functions of error are often used to mitigate these problems. The steady state error can be mitigated by integrating the discrepancy into a stock, then using the integrated value to additionally affect policy. This is called integral control. The sluggish adjustment can be mitigated by additionally driving policy by a derivative of the discrepancy. And this is called derivative control.

A combination of the three type functions of the discrepancy is called PID (proportional, integral, and derivative) control in engineering and is widely used in servomechanisms. The process is transferrable to managerial, economic, and other organizational contexts when simple goal seeking policies are inadequate. Supply chains, and meeting developmental targets are cases in point that we'll discuss later in the policy context.

5. Co-flow as a non-policy occurrence

In principle, a policy driving a flow entails a decision process using available information. Since the instantaneous value of a flow is never available as all flows must be observed for some time before a perception can be formed about their magnitude, a flow cannot be used in a model to drive another flow. However, in a complex system consisting of multiple conservative subsystems, some flows may occur concomitantly even though they are connected to different stocks. For example, a shipment out of inventory happens simultaneously with an equivalent reduction in order backlog and an increase in the cash register by the value of the shipment. Likewise, the separation of a skilled worker will occur simultaneously with the reduction in the skill level of the workforce and a decrease in payroll. Such a dependence of a flow in one subsystem on one another is not based on a policy but arises out of an intrinsic relationship between the two as shown in Figure 3.7 where a primary stock may have a tracking stock in another subsystem. A perceived value of a co-flow may however form basis for a policy. For example, capital retirements may drive capital formation modified by other factors; and attrition rate of workers may drive a recruitment policy along with other considerations. These will be discussed in the section on complex policies.



*coflow = primary flow*conversion factor*

Figure 3.7 Co-flow structure

The change in the tracking stock in such cases can be represented by multiplying the primary flow by a conversion factor that appropriately transforms units of the primary flow to match those of the tracking stock. Following examples illustrate how co-flow structure can be used to transform primary flows into tracking flows:

$ value of sales = Sales * price charged per unit

Increase in salary budget = New hires*salary per hire

Decrease in backlog = Shipments * fraction of orders filled per shipment

Increase in total weight of employees = New hires * average weight per new hire

Note, in all equations representing policy, the units on both sides of the equation must balance. Also, a conversion factor must be a discernable parameter established independently of the model. All decisions are of course modulated by the personality factors subsumed in the parameters designating the aggressiveness of action or the speed of the correction it creates.

6. Perception and expectation formation as drivers of managerial decisions

The information used in the decision process can often not be directly inferred from the system it is used in. Information sources must be observed for some time to perceive their magnitude, which is often a weighted average of several past observations rather than an instantaneous value. Especially, when a flow factors into a decision, instantaneous value cannot even be measured, and its perception must arise from observations over some length of time.

Forrester represented perception, that is the first step in expectation formation, as a moving exponential average of information. He also suggested setting up the macros for perceiving information differently from the delay process entailed in a supply chain. Hence, two macros were created for use in DYNAMO. SMOOTH to represent information perception, DELAY to represent supply-line lags, which will be discussed elsewhere.

Following the structure of DYNAMO, the averaging of information or smoothing (SMTHN) is set up as an exponential moving average in Stella, N standing for the order of the process. Thus, the first order exponential moving average SMTH1 has the structure shown in Figure 3.8:

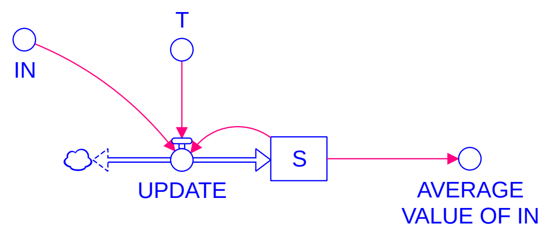


Figure 3.8 **Stock and flow diagram of SMTH1 macro.**

Following equations correspond to this system:

S(t) = S (t - dt) + (IN-S) * dt/T

INIT S = IN

Where IN is the input to the macro and S is its output.

It can be shown experimentally that at any point in time, the stock will contain an exponential average of the various vintages of input, most recent vintage being weighted most and the weights exponentially decreasing with vintage.

Before I describe the experiment, for easier comprehension, let me first split the flow into two parts. Figure 3.9 shows the model with this transformation.

The corresponding model equations after this transformation can be written as follows:

S(t) = S (t - dt) + (INFLOW - OUTFLOW) * dt

INIT S = 0

INFLOW = IN/T

OUTFLOW = S/T

AVERAGE VALUE OF IN = S

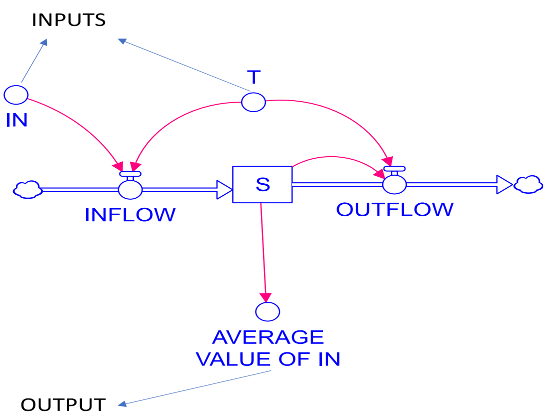


Figure 3.9 **SMTH1 Macro with updating flow split into inflow and outflow.**

Now if we inserted a single pulse with amplitude 1 unit into this system, the corresponding values of stock, IN, S and outflow can be calculated as in Table 1.

Table 1 **Calculation of pulse response of SMTH1 macro**

TIME	IN	S	OUTFLOW
0	0	0	0
1	1	1	0
2	0	1-1/T	1/T
3	0	$(1-1/T)^2$	(1-1/T)/T
4	0	$(1-1/T)^3$	$(1-1/T)^2/T$
5	0	$(1-1/T)^4$	$(1-1/T)^3/T$
6	0	$(1-1/T)^5$	$(1-1/T)^4/T$

Note that the weight of the input remaining in the stock decreases exponentially with time. If there is a regular input stream instead of a one-time pulse, the stock at any time will have a collection of vintages, with the weight of older vintages declining exponentially with time.

We can also determine the weighting function mathematically as follows:

$$S_t = S_{t-dt} + dt/T\,(IN_{t-dt} - S_{t-dt})$$

For simplification, let DT=1 and T >> 1 to avoid integration error.

$$S_t = IN_{t-1}/T + S_{t-1}\,(1-1/T)$$

$$S_{t-1} = IN_{t-2}/T + S_{t-2}\,(1-1/T)$$

Substitute value of S_{t-1} in the equation for S_t

$$S_t = 1/T\,(IN_{t-1} + IN_{t-2}(1-1/T)) + S_{t-2}(1-1/T)^2$$

$S_{t-2} \quad = \quad IN_{t-3}/T + S_{t-3}(1-1/T)$

$S_{t-3} \quad = \quad IN_{t-4}/T + S_{t-4}(1-1/T)$

……….

If we continue to substitute values of S_{t-x}, we get,

$S_t \quad = 1/T\ ((IN_{t-1} + IN_{t-2}(1-1/T) + IN_{t-3}(1-1/T)^2) + IN_{t-4}(1-1/T)^3 + \ldots\ldots.) + S_0(1-1/T)^n$

$S_0(1-1/T)^n$ tends to 0

$S_t \quad = 1/T\ ((IN_{t-1} + IN_{t-2}(1-1/T) + IN_{t-3}(1-1/T)^2 + IN_{t-4}(1-1/T)^3 + \ldots\ldots.)$

which is similar to the weighting function of first order smooth we arrived at through experiment route. Higher order delays and averaging processes can create complex weighting functions. Calculating these complex functions mathematically is quite cumbersome. These weighting functions can however be easily determined experimentally.

Order of smoothing and the corresponding weighting functions of the exponential average

Both experimental and mathematical determinations of the weighting functions of the information in the stock for 1st order smoothing have an exponential shape, indicating that recent information would be weighted most heavily while the weights of the vintages decrease exponentially. We can find the weighting functions of higher order smoothing experimentally using simulation once we have created the algorithm. Figure 3.10 shows the response of the various orders of the smoothing macro SMTHN in Stella using a pulse of one unit as input. Please note, the input is divided by DT in the macro, so Pulse height applied over the period DT=1/DT while the area of the rectangle created by the pulse =1.

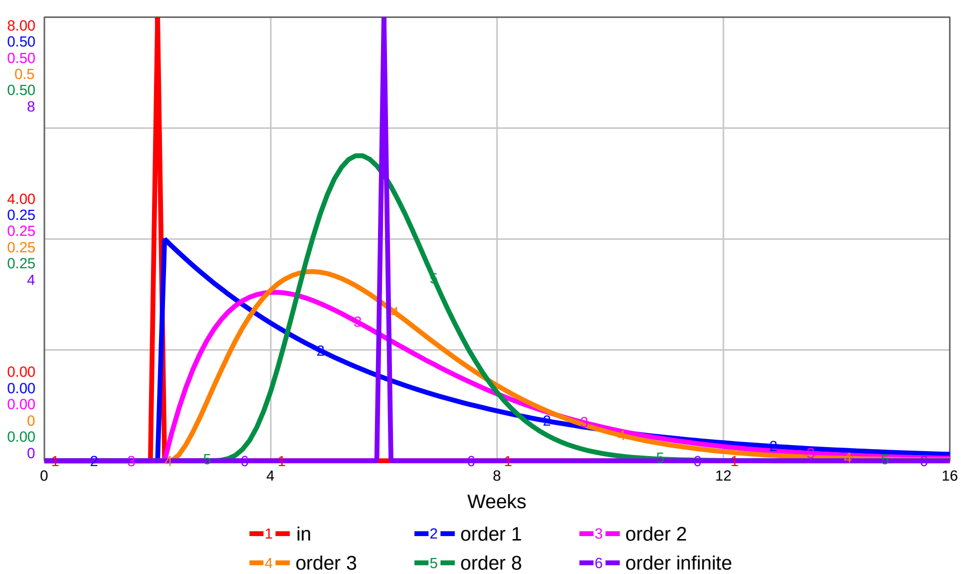


Figure 3.10 Weighting functions of exponential moving averages of various orders

As the order of the averaging process increases, the weights of the recent vintages are reduced while those of the midrange vintages rise. Forrester most frequently used 1st and 3rd order smoothing in his models. His position was that these have fundamentally different weighting functions that approximate perception, the former weighing recent information most heavily and the latter giving more importance to midrange information. Thus, all types of perception can be approximated by 1st and 3rd order smoothing functions. An infinite order smooth will replicate the input after an almost discrete delay, which can be created by the software, but perhaps should never be used as it cannot be related to any form of human perception.

Use of forecasts in forming expectations

Expectations are often directly based on the perceptions created by the averaging processes discussed earlier in this section. Managers sometimes use a forecast of future values based on the trends created by past observed values. This calculation involves using current and past values of the variable to compute slope of the change and then project it over desired period of time as explained by Forester in Appendix L of Industrial Dynamics (6).

Most modeling software includes a built-in macro for forecasting that performs trend extrapolation. Inputs to the forecast include the value of input, the averaging time for input to calculate its past value, and the distance into future. For example, Stella software used in this chapter may forecast future sales using FORCST built-in macro as follow:

Sales Forecast = FORCST (Sales, Sales averaging time, forecast period, initial growth trend)

The macro produces a forecast of sales “forecast period” into the future. The forecast is based on current sales, and the trend in sales over the sales averaging time. The initial growth trend in sales can be set to 0 if there initially was no trend.

Use of forecasts in policy can be destabilizing when turning points exist in the profile of the variable being forecast. Use of *ex post error* as done in PID control is therefore a more reliable way to align a policy with management goals (12).

Combining generic decision rules for representing complex policies

In practice, the six generic components of policy discussed above are combined by managers in appropriate ways to device more complex policies that are used to make repetitive decisions. The decisions then create flows that change information to be used in the subsequent decisions. Following example illustrate some of such complex policies.

Capacity additions

Capacity additions may depend on the perceived value of capital retirements, modified by a multitude of factors such as interest rate, perception of capacity access or shortage, expected demand, etc. Thus, as shown in Figure 3.11, capacity additions policy will combine a co-flow of retirements, and an influence from a perception of capacity adequacy, which may arise from an assessment of future demand through recognition of changes in own backlog of orders and possibly also a forecast of future demand – all leading to the creation of an implicit stock adjustment process. Additionally, capacity additions may also be affected by the cost of capital manifest in the current interest rate, which although seemingly exogenous, but would in the long run depend on velocity of money that is influenced by the production capacity.

While the managers might normally attempt to replace the retired capacity in equilibrium, they modify the replacement rate by a multitude of implicit stock adjustment processes requiring use of perception and formation of expectations in other situations. They might also factor in the cost of capital manifest in the interest rate, reducing capacity addition when interest rates are high and increasing it when they are low. Thus, capacity additions policy in Figure 3.11 can be stated as:

capacity additions = SMTH1(capacity retirements, time to perceive capacity retirements) * effect of capacity adequacy * effect of interest rate

whereas the effect of capacity adequacy is a GRAPH of production capacity/desired capacity

and, the desired capacity depends on expected future demand, which is a forecast of order backlog.

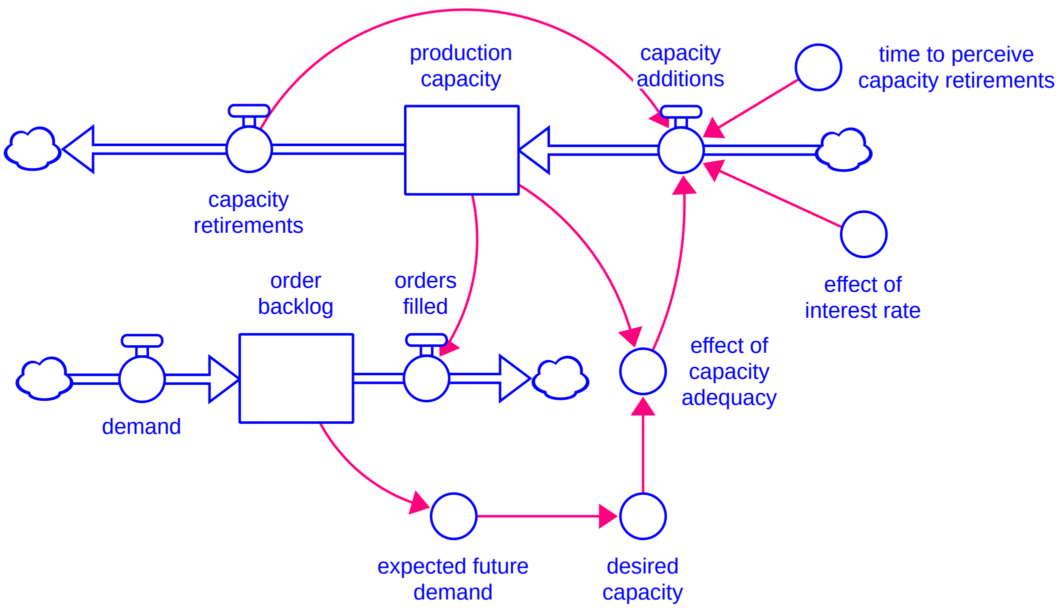


Figure 3.11 combining co-flow and implicit stock adjustment rules in the complex capacity addition policy.

Worker hiring

Likewise, worker hiring policy may at the outset attempt to replace attritions but with modifications incorporating labor market considerations and workforce adequacy as illustrated in Figure 3.12.

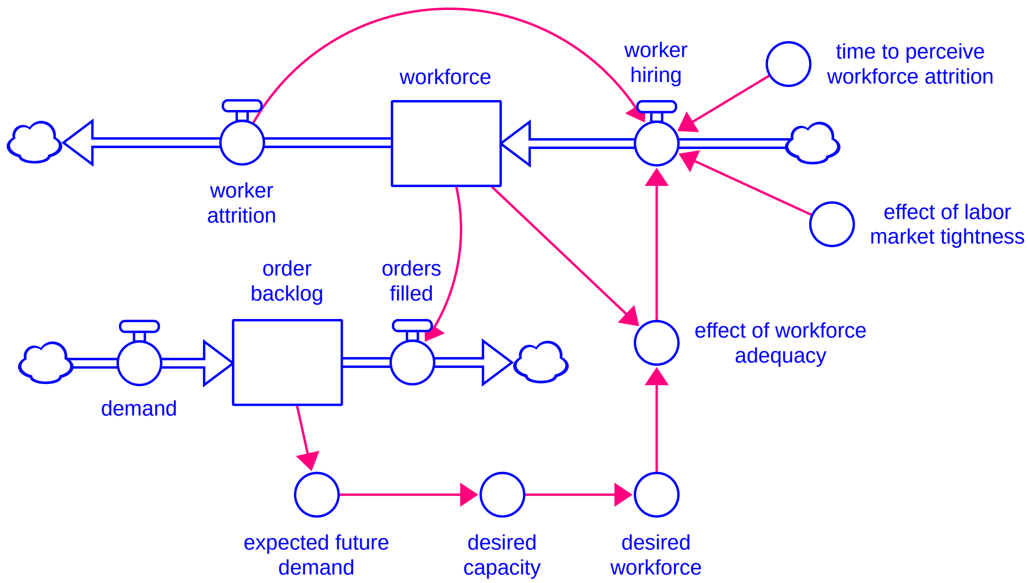


Figure 3.12 combining co-flow and implicit stock adjustment rules in the complex worker hiring policy.

Thus, worker hiring might be determined by combining perception of attritions – represented by the smooth of a co-flow, modified by the implicit stock adjustment processes manifest in the effects of workforce adequacy and labor market tightness as in the equation below:

Worker hiring = SMTH1(worker attrition, time to perceive worker attrition) * effect of workforce adequacy * effect of labor market tightness

Growth of a species

The growth of a species might depend on the compounding processes affecting both births and deaths modified by the implicit stock adjustment processes manifest in effects of food availability and pollution, both arising out of the species population as illustrated in Figure 3.13.

Thus, species births = species population * Effect of food availability * effect of pollution

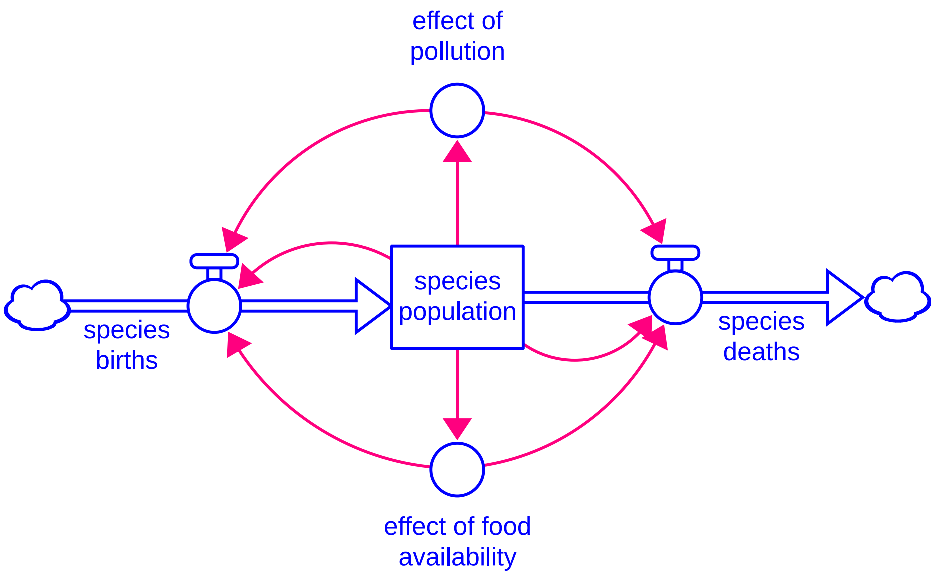


Figure 3.13 combining compounding and implicit stock adjustment in representing births and deaths in the growth of a species

Plane seat additions

Plane seat addition policy in an airline might be driven by the revenue generated by existing plane seats, a co-flow process, modified by financial constraints – an implicit stock adjustment process as illustrated in Figure 3.14. Thus,

Seat additions = revenue * fraction of revenue allocated to seat additions * financial constraint

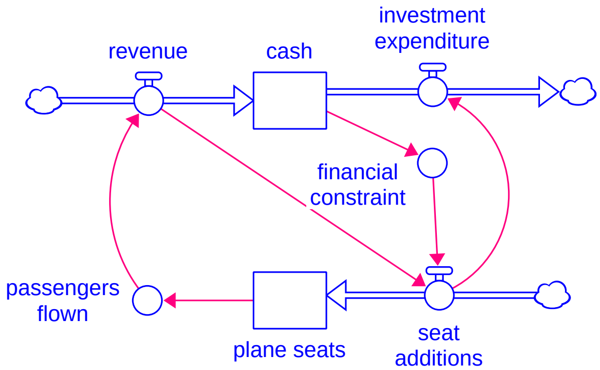


Figure 3.14 Combining co-flow and implicit stock adjustment in plane seat addition policy.

These are only a few of the examples of how most complex managerial decisions constitute combinations of simple generic rules. The long-term outcome of decisions cannot be accurately predicted, but the outcomes resulting from one set drive subsequent sets. The iterative process arising out of this regimen can lead to optimal long-term outcomes. The managerial decision process also creates feedback between subsystems multiple managers might manage. This feedback leads to complex pattens of behavior subsuming growth, decline, overshoot, and oscillations, which are discussed in the next section as ubiquitous systems existing across disciplines.

Generic systems created by managerial role play

At the outset, it should be recognized that complex behavioral patterns are created by coupled feedback loops formed by repetitive managerial actions following intuitive policies discussed in the last section. Table 2 summarizes how various types of feedback loops contribute to system behavior.

Table 2 Feedback loop types and their behavior

	Contribution to system behavior	
Feedback loop type	*Minor*	*Major*
Positive	Growth/decline snowball	Growth/decline snowball
Negative	Stable goal seeking	Unstable goal seeking/Oscillation

Positive feedback loops create snowball effect resulting in either exponential growth or exponential decline, while negative feedback loops drive goal seeking behavior either following a stable or a cyclical path. Minor and major labels refer to the order of the feedback loop specifying the number of stocks in it. All feedback loops must have at least one stock in them. Minor loops always have one stock, while major loops have more than one. This distinction is important as the behavior of negative feedback with one stock substantially differs from one with more than one stock. The former creates a stable path to a goal, while the later may seek a goal through an unstable path, creating cyclical tendencies. A combination of these loops can create many complex patterns including growth, goal-seeking behavior, and various oscillatory modes. We will discuss the computable structure of basic systems creating growth, limits to growth, goal seeking behavior, and oscillations.

a) Growth systems

We will discuss 4 types of growth patterns and their underlying structure.

1. Unconstrained growth that has exponential shape. It can be caused both by minor and major positive or reinforcing feedback loops and is driven by what is called the loop gain in engineering.
2. S-shaped behavior that initially exhibits exponential growth but moves towards a plateau as its gain varies in a non-linear fashion through a coupled minor negative feedback loop.
3. Overshoot and oscillation, which is created by the structure that generates S-shape with a delay in it that increases the order of the negative feedback loop.

4. Overshoot and sustained decline, which is caused by growth additionally draining its sustaining stock.

1. Unconstrained growth

Unconstrained growth is often exponential in shape. It is caused by positive feedback, which creates snowball effect. The growth behavior of a positive FB loop can be inferred from its gain. Normally, whether major or minor, a positive FB loops will exgibit either growth or decline in a snowball pattern. The rate of this snowball effect will depend on what is called an open loop gain or OLG for a minor loop and an open loop steady state gain or OLSSG for a major loop.

The gain of a minor feedback loop can be computed by breaking the information link between the stock and the flow and treating the arrowhead of the link as input to an open system so created and the tail end as output as illustrated in Figure 3.15 for a simple population growth system. The ratio of output Po to input Pi is the open loop gain OLG that determines the nature of growth.

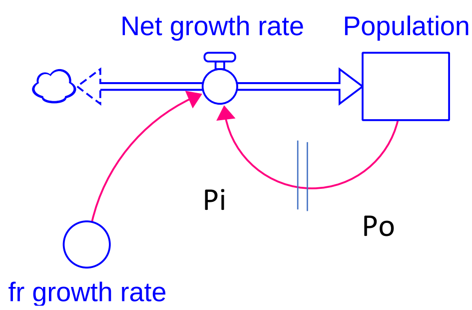


Figure 3.15 Example of a minor positive feedback loop

Po= Pi*(1+fr growth rate)

OLG Po/Pi = (1+fr growth rate)

If the OLG exceeds 1, the system will exhibit exponential growth. If it is less than one (realized when fr growth rate is negative), exponential decline happens. OLG of 1 yields steady state.

The gain of a major positive feedback loop is computed similarly, except the loop must be broken in the information path connecting the multiple conservative systems (stock-flow structures) rather than the stock and the flow in a single conservative system. As shown in Figure 3.16 for a resource-driven population growth process, this break creates an open system with output = Po and input = Pi. Gain can be computed as the ratio between Po and Pi. The computation assumes there is steady state in each conservative system in the loop. Hence the name Open Loop Steady State Gain (OLSSG).

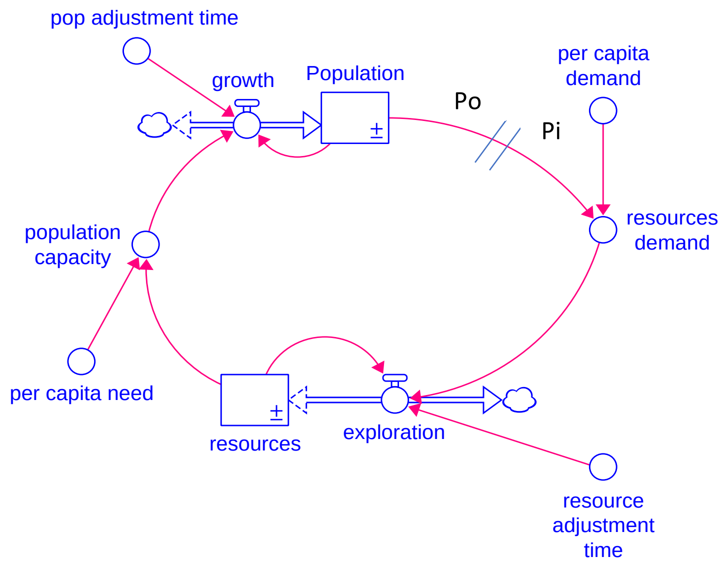


Figure 3.16 Calculating OLSSG of a major positive FB loop

Po = population capacity

= Resources/res per capita need

= Resources demand/res per capita need

= Pi* resources per capita demand/res per capita need

OLSSG Po/Pi = resources per capita demand/res per capita need

The calculation of gain in this way can quickly point to the policy parameters that may be targeted for intervention if the growth pattern must be changed.

2. *S-shaped growth*

When growth occurs in the short run by a compounding process but is constrained by capacity limits, an S-shaped pattern might be experienced. The capacity limits can be imposed by fixed resources or created endogenously by the growth itself through entropic processes such as increase in organizational complexity, crowding, bureaucratization, etc. Such a pattern is created by coupled positive and negative feedback loops. Figure 3.17 shows the structure underlying S-shaped growth created by a capacity limit using another variation on the population growth process in which population growth is initially driven by the minor loop arising out of a compounding process, but whose gain declines due to resource constraints.

Essential equations for the model of Figure 3.17 are:

Population(t) = Population (t - dt) + (Net growth rate) * dt
Net growth rate = Population * growth fraction
Growth fraction = GRAPH (resource availability)
resource availability = per capita resources/resources per capita need

per capita resources = resources/population

For the simplest case, we can assume that the resources stock does not change, and per capita need is also constant.

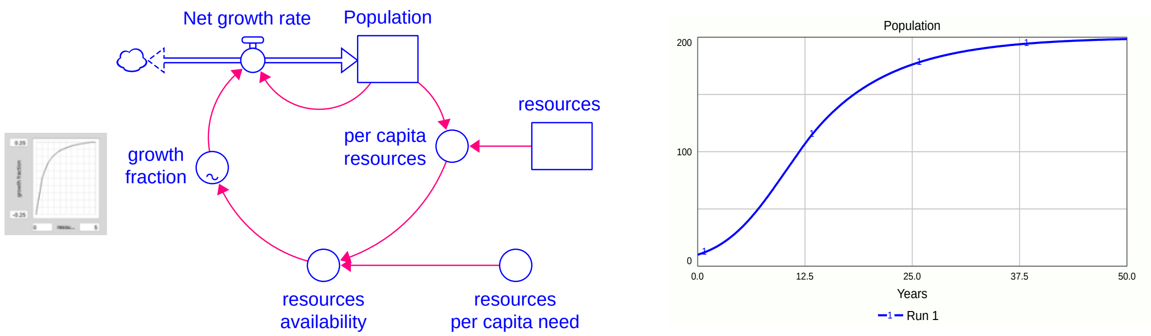


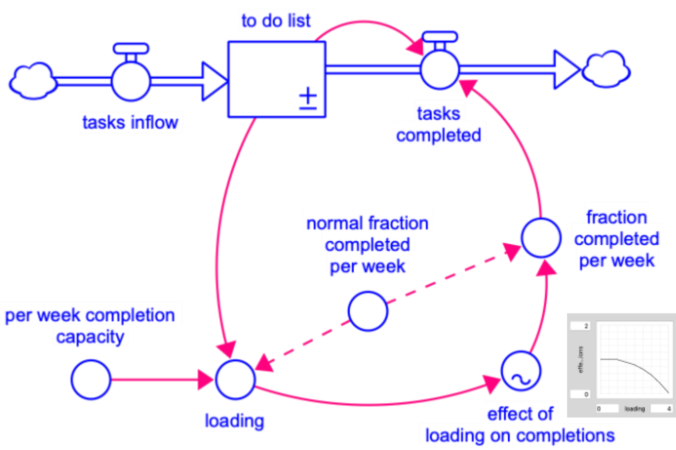
Figure 3.17 S-shaped growth arising out of capacity constraints.

The S-shaped behavior arising from this computable structure is often explained as an outcome of changing loop dominance when a positive and a negative feedback loop are coupled through a nonlinear relationship. Over the course of exponential growth part of the graph, the positive feedback loop dominates behavior; over the asymptotic growth part, the negative feedback loop is dominant. This explanation, although correct is a bit abstract. In fact, while the growth in this system is driven by the gain of the positive feedback loop, this gain is not constant but is varied by the coupled negative feedback loop whose impact depends on resource availability. With constant resources and increasing population, the resource availability declines, which reduces the gain of the growth process until net growth rate is driven down to zero. The constant resource stock thus acts as the capacity constraint.

In many instances, a capacity constraint might also arise out of growth itself, creating an endogenous limit. Examples of this include increase in travel time due to congestion even though more roads are built, increase in disagreements from a growth in workforce that may limit productivity, increase in pollution from a rising population that may limit its growth, increase in defects and rework from a rising production capacity that may limit output, increase in delivery delay from rising sales, that might limit sales, etc. These endogenously created problems often grow geometrically even when the growth is linear. Inability to address them will undermine growth. Attempts to create infrastructure to address them will draw down resources that could otherwise fuel growth. Either way, an endogenous limit created by growth can limit it.

3. *Tipping point*

Like the system creating S-shaped growth, a tipping point system also arises of coupled negative and positive feedback loops, but with the negative loop dominating in the short run while the positive feedback assumes dominating role after a tipping point. Examples of such dominance shift appear in many physical, environmental and human systems. A structural beam will bend at first in response to increasing load but break when the load exceeds the restraining forces created by its elasticity. Fish regeneration in a fishery might at first increase with catch as the lowering of fish population increases munificence for the remaining fish but decrease when catch rises beyond the ability of the fish to regenerate. Increasing CO_2 emissions invokes increasing removals by nature until its absorption capacity is over-loaded, in which case removals would decline and CO2 concentration would rise uncontrollably. People are often able to cope with increased workload by working harder but may experience burnout due to fatigue limiting their productivity when they are loaded beyond their capacity. All such systems are driven by a control mechanism that eventually yields to a short run growth process. Figure 3.17(a) shows the structure of a tipping point system using worker burnout as a metaphor.



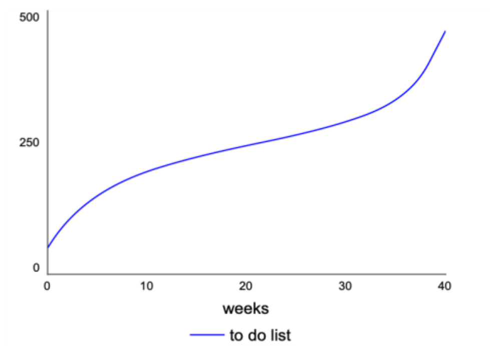


Figure 3.17(a) Tipping point arising out of capacity constraints

Essential equations for the model of Figure 3.17(a) are:

to do list(t) = to do list(t - dt) + (tasks inflow - tasks completed) * dt
tasks completed = to do list*fraction completed per week
effect of loading on completions = GRAPH(loading)
fraction completed per week = normal fraction completed per week*effect of loading on completions
loading = (to do list*normal fraction completed per week)/per week completion capacity

When tasks inflow rises, working harder initially allows a worker to increase task completions, which depletes to-do-list. This process will seek equilibrium with a higher to do list. However, when tasks inflow exceeds a certain threshold, persistent hard work above capacity will create fatigue that will lower productivity, which will reduce task completions, thus causing uncontrolled growth in to do list. Policy interventions in a tipping point system are often path dependent, meaning an early intervention will require less effort to reinstate a system than a later intervention.

4. Overshoot and oscillation

When a delay is inserted in the negative feedback loop controlling the gain parameter as illustrated in Figure 3.18 that builds on the population growth model of Figure 3.17, overshoot and oscillation happens.

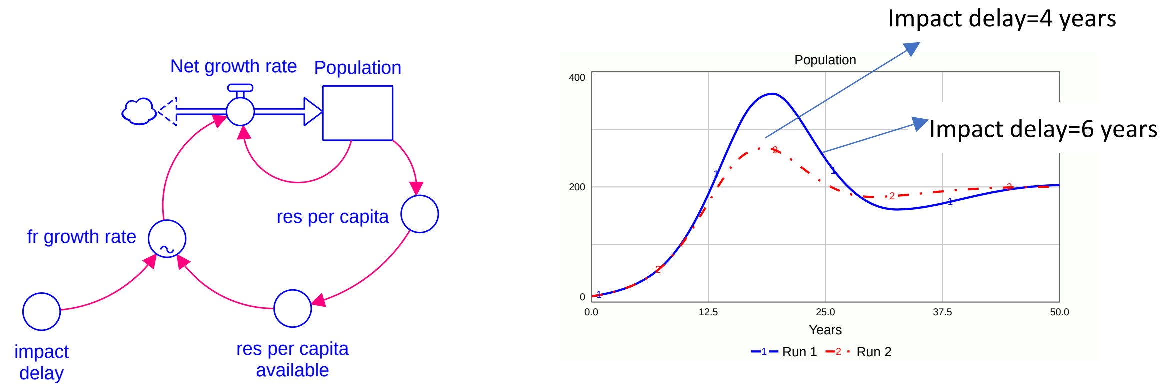


Figure 3.18 Overshoot and oscillation arising from delayed response to capacity constraint

A longer delay increases the amplitude of the overshoot. The delay defers the constraining of the fractional growth rate until after the population stock has exceeded its equilibrium value. Likewise, when reduction in growth rate finally kicks in, it continues beyond the equilibrium value of the stock. The delay creates structure for a second order negative feedback loop that is inherently unstable. Combined with the positive feedback driving growth, the outcome portrays both growth and oscillation.

5. *Overshoot and sustained decline*

Using another variation on the population growth process illustrated in Figure 3.19, it can be shown that when the growing stock of population drains the sustaining stock of resources, an overshoot followed by a sustained decline happens, since growth depletes its sustaining stock.

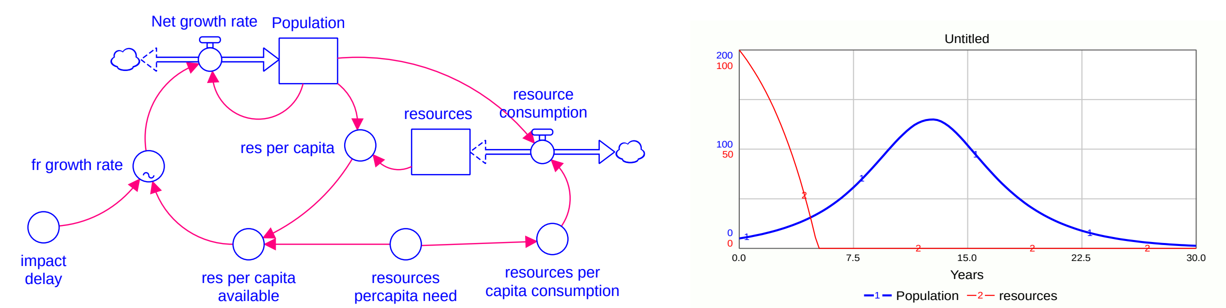


Figure 3.19 Overshoot and sustained decline arising from capacity drain caused by growth

b) Goal seeking behavior and oscillation from negative feedback

Goal seeking behavior is asymptotic in nature. It is driven by a minor negative feedback loop. Oscillations on the other hand are created by major negative feedback. The generic systems underlying the various forms of goal-seeking behavior created by coupled negative feedback loops are described below:

1. Goal seeking behavior

Goal seeking behavior is created by the first order stock adjustment process, explicit as well as implicit, as we discussed in the context of Managerial Role Play. Figure 3.20 applies this process to population with fixed resources.

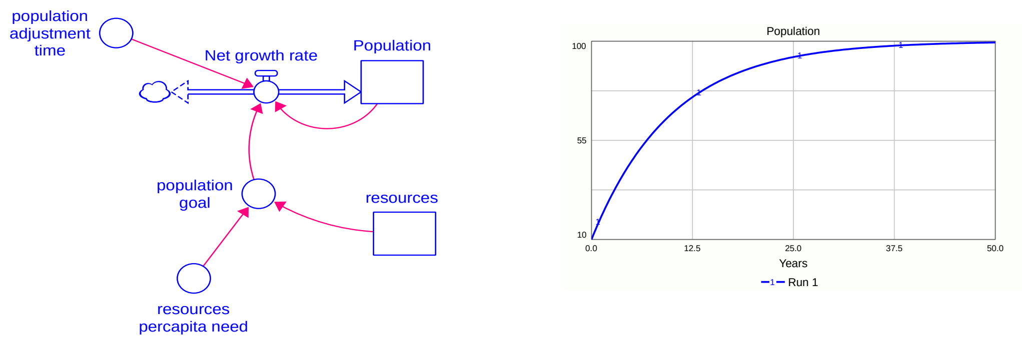


Figure 3.20 Goal seeking behavior fom stock adjustment process

While the goal for the population is created by external resources, the discrepancy between population goal and existing population is adjusted iteratively, becoming smaller in each successive iteration, which gives rise to an asymptotic growth pattern.

2. Oscillatory behavior

Oscillatory behavior arises out of major negative feedback. When multiple stocks are connected through an information path forming negative feedback, the time of equilibrium in the feedback loop must coincide with time of equilibrium in each stock in the loop, which is hard to achieve. Hence when one stock meets the conditions of equilibrium, others are often out of equilibrium; and their adjustment disturbs equilibrium in the first stock. A major negative feedback loop thus may seek an equilibrium but never achieve it, creating an oscillatory pattern of behavior. An example of a hypothetical frictionless mass-spring system that can oscillate forever is shown in Figure 3.21

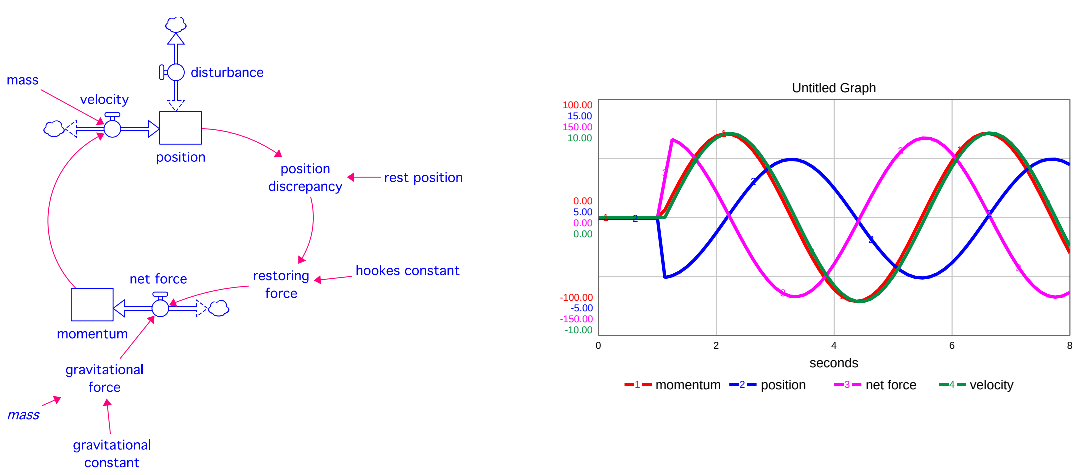


Figure 3.21 Oscillatory behavior in a mass-spring system

The mass spring system consists of two stocks, position and momentum, each driving the adjustment of the other. The system can be represented by the following two integrations portrayed in Figure 3.21:

Position = ∫ (velocity – disturbance) * dt
    Velocity = Momentum/mass
Momentum = ∫ (grav. force – restoring force) * dt
    grav. force = mass*g
    restoring force = displacement* Hooke's constant
    displacement = rest position – position

When disturbed using a one-time change, this system will break into oscillations as Position and momentum attempt to restore equilibrium conditions in each other.

The structure of the mass-spring system can be mapped into a simple supply chain consisting of inventory and workforce stocks as shown in Figure 3.22 with its quintessential instability.

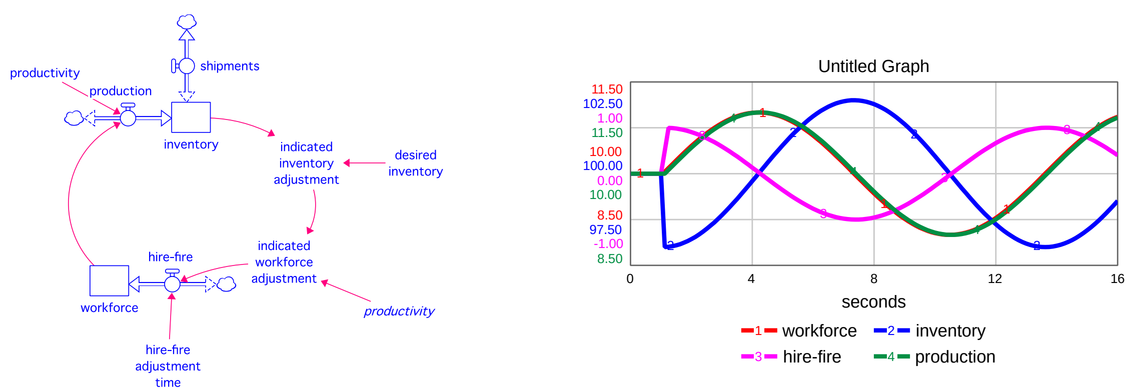


Figure 3.22 Structure of mass-spring system mapped into a simple supply chain

The corresponding integrations would be:

Inventory = ∫ (production – shipment) *dt
    production = workforce * productivity
workforce = ∫ (hire-fire) * dt
    hire-fire = indicated workforce adjustment/hire-fire adjustment time
    indicated workforce adjustment = indicated inventory adjustment/productivity
    indicated inventory adjustment = desired inventory – inventory

Introduction of a perception lag in this system will change the order of the negative feedback loop to 3 and create exploding oscillations as shown in Figure 3.23. The perception lag will change the system equations as follows:

    indicated workforce adjustment = perceived inventory adjustment/productivity perceived inventory adjustment = SMTH1(indicated workforce adjustment, perception lag)

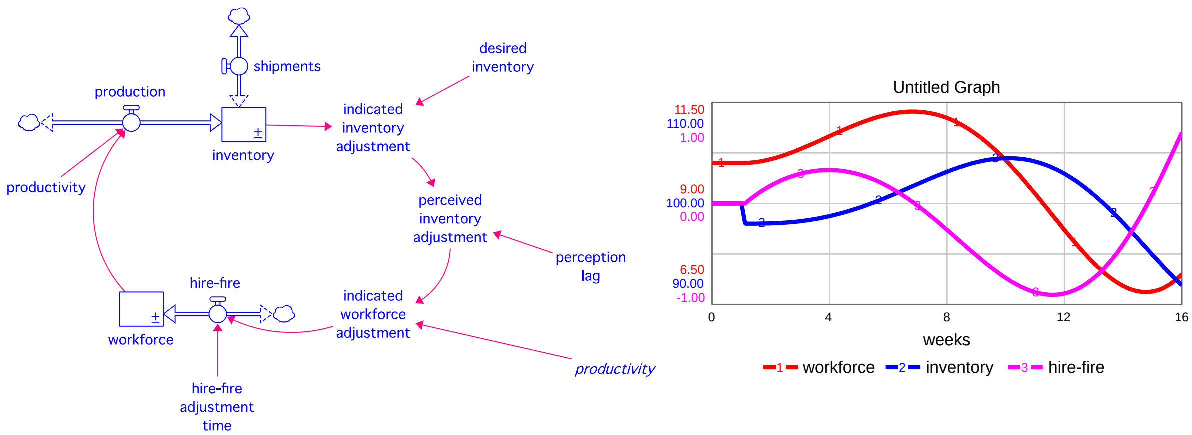
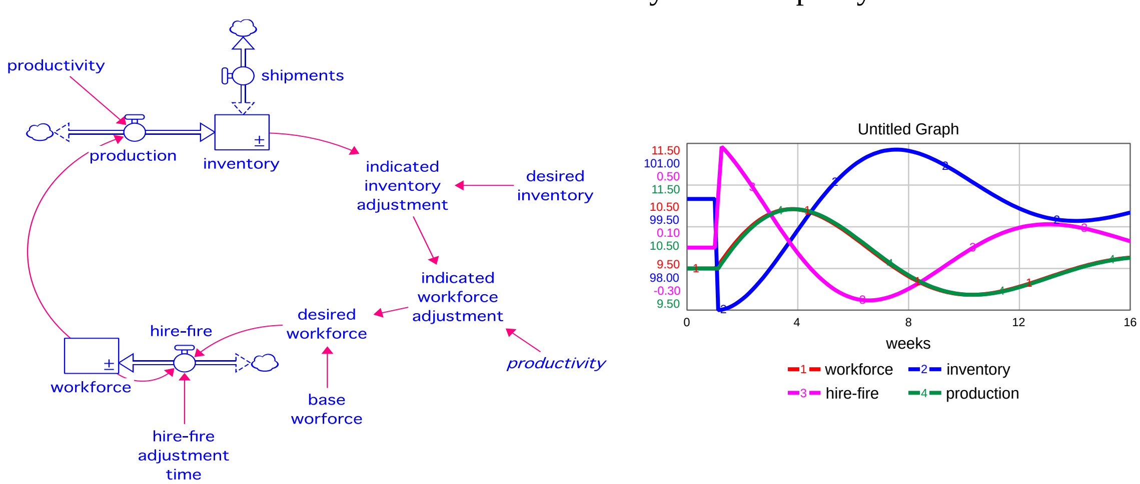


Figure 3.23 Introduction of a perception lag into simple supply chain exacerbating oscillation

While the oscillatory character of the major negative feedback loop persists, the lag further increases the time difference between adjustment of the stocks of Figure 3.22, that exacerbates the overshoot in each successive cycle.

Another variation on the system of Figure 3.22 is shown in Figure 3.24 where indicated workforce adjustment is added to a base level workforce to obtain the desired level of workforce that must be achieved. hire-fire is then determined by the discrepancy between the desired and



the current levels of workforce.

Figure 3.24 Damping created by adding minor feedback to the system of Figure 3.22

This variation modifies system equations as follows:

hire-fire = (desired workforce – workforce)/hire-fire adjustment time desired workforce – base workforce + indicated workforce adjustment

This change adds a minor feedback loop to the system arising from the workforce adjustment process, which augments adjustment in each successive cycle, hence creating damping in this system.

Variations on coupled negative feedback loops and their behavior as well the contingencies under which goal seeking behavior and oscillation may originate from coupled positive feedback are discussed elsewhere.

Summary

This chapter described generic processes driving managerial policy and the basic systems they form. While contemporary economic theory likens firms and markets to rational agents working with perfect information for mathematical convenience, they in reality consist of multiple managers working with limited information to meet their short-term goals. System thinking calls for modeling their bounded rational role-play. The bounded rational policies managers use are not based on random logic but are a combination of a small number of decision rules. These policies may also not be denoted as logic statements as if driving a switching process in a digital computer but must be represented as continuous mathematical relationships subsuming decision patterns. Understanding these rules, their combinations, and the generic systems they create are key to creating realistic models of economic systems.

Everyday policies of managers call for meeting goals expected of them by combining the information available at the decision points, often not knowing the consequences of their decisions. For modeling purposes, the decisions can be broken down into simple rules managers use to meet every day demands of their roles, which include production, compounding, draining, and adjustment towards an explicit or implicit goal. Those policies may also involve perception formation and forecasting. They are combined in various configurations to meet the demands of a complex role. Managerial policies drive flows in the system and are driven by the information contained in stocks and parameters. The decision process may additionally create concomitant flows in interacting conservative systems, which are designated as co-flows in a model. Managerial role play creates complex behavior patterns that can also be decomposed to simpler parts like growth, overshoot, goal seeking behavior and oscillation. Understanding the managerial decision process and the basic patterns of behavior it creates in economic systems is key to addressing operational policy design that can be implemented through modifying managerial roles.

References

1. Katz D, Kahn RL. The social psychology of organizations. 2d ed. New York: Wiley; 1978. vi, 838 p. p.
2. Simon HA. Models of bounded rationality. Cambridge, Mass.: MIT Press; 1982. v. <1-3 > p.
3. Saeed K. Designing an Environmental Mitigation Banking Institution for Linking the Size of Economic Activity to Environmental Capacity. Journal of Economic Issues. 2004;38(4):909-37.
4. John DWM. Rationality in the Analysis of Behavioral Simulation Models. Management Science. 1985;31(7):900-16.

5. Forrester JW. Principles of systems : text and workbook. 2nd preliminary ed. Cambridge: Available from Technology Store, M.I.T. Student Center; 1968. v. p.
6. Forrester JW. Industrial dynamics. Cambridge, Mass.: M.I.T. Press; 1961. 464 p. p.
7. Forrester JW. Urban dynamics. Cambridge, Mass.,: M.I.T. Press; 1969. xiii, 285 p. p.
8. Saeed K. Urban dynamics: A systems thinking framework for economic development and planning. . ISOCARP Review. 2015;11:129-32.
9. Saeed K. Dynamics of Income Distribution in a Market Economy: Possibilities for Poverty Alleviation. In: Dangerfield B, editor. System Dynamics: Theory and Applications. New York, NY: Springer US; 2020. p. 491-522.
10. Saeed K. The dynamics of economic growth and political instability in developing countries. System Dynamics Review. 1986;2(1):20-35.
11. Saeed K, Pavlov OV, Skorinko J, Smith A. Farmers, bandits and soldiers: a generic system for addressing peace agendas. System Dynamics Review. 2013;29(4):237-52.
12. Saeed K. Can trend forecasting improve stability in supply chains? A response to Forrester's challenge in Appendix L of Industrial Dynamics. System Dynamics Review. 2009;25(1):63-78.