5.12 SYSTEM DYNAMICS: SYSTEMIC FEEDBACK MODELING FOR POLICY ANALYSIS

Yaman Barlas

Industrial Engineering Department, Boğaziçi University, Istanbul, Turkey

CONTENTS

"Que sais-je?" ("What do I know?") (Montaigne)

To the memory of my father who somehow taught me to be a student for life. . .

SUMMARY

The world is facing a wide range of increasingly complex, dynamic problems in the public and private arenas alike. System dynamics discipline is an attempt to address such dynamic, long-term policy problems. Applications cover a very wide spectrum, including national economic problems, supply chains, project management, educational problems, energy systems, sustainable development, politics, psychology, medical sciences, health care, and many other areas. This article provides a comprehensive overview of system dynamics methodology, including its conceptual/philosophical framework, as well as the technical aspects of modeling and analysis. The article frequently refers to other articles in the same Theme as appropriate. Today, interest in system dynamics and related systemic disciplines is growing very fast. System dynamics can address the fundamental structural causes of the long-term dynamic contemporary socio-economic problems. Its "systems" perspective challenges the barriers that separate disciplines. The interdisciplinary and systemic approach of system dynamics could be critical in dealing with the increasingly complex problems of our modern world in this new century.

1. INTRODUCTION

At the start of the new century, the world is facing a wide range of increasingly complex, dynamic problems in the public and private arenas alike: nations desire economic growth, yet growth also results in environmental and ecological – sometimes irreversible – destruction. Chronic high inflation, budget deficits, and simultaneous high unemployment together constitute a persistent problem for many developing countries. There is a widening gap between the so-called "north" and "south" nations and also a widening gap between the poor and the rich in a given nation. Markets – both commodities and financial – generate all sorts of short, medium, and long-term cycles, the uncertainty of which has been a major problem for private and public decision-makers for decades. Thousands of small companies are initiated each year, many enjoying a fast – yet unbalanced – growth, only to be followed by bankruptcy in a few years. Globalization is posing new challenges in the social, economic, and corporate domains. With the unprecedented speed of international communication, the "new" economy means that companies will soon find themselves in a complex network of relationships and only those that can understand its dynamics will be able to compete. We are already experiencing the first worldwide recession of the "new" economy – of the new millennium. In the socio-economic domain, there is a tension between the interests of the nation-states and those of international conglomerates, between increasing international interaction and increasing micro-nationalism. Many nations worldwide face the dilemma between full democracy/human rights and the special measures often needed against terrorism.

The examples listed above have some common defining characteristics: they are all dynamic, long-term policy problems. "Dynamic" means, "changing over time." Dynamic problems necessitate dynamic, continuous managerial action. Optimum oil well location problem may be a very difficult problem, but it is nevertheless a "static" decision problem. The decision is made once, and it is not periodically monitored and adjusted depending on the results. The dynamic problems mentioned above, however, must be continuously managed and monitored. Thus, in the specific context of management and policy making, dynamic problems are the ones that are of persistent, chronic, and recurring nature. We take managerial actions, observe the results, evaluate them and take new actions, yielding new results, observations, further actions, and so on, which constitutes a "closed loop." In other words, most dynamic management problems are "feedback" problems. Feedback loops exist not only between the control action and the system, but also in between the various components within the system. Therefore, it is also said that such dynamic feedback problems are "systemic" in nature, that is, they originate as a result of the complicated interactions between the system variables. Finally, since dynamic management necessitates a stream of dynamic decisions, the research focus should not be the individual decisions, but the rules by which these decisions are made, that is, the "policies." The individual decisions are the outcomes of the application of the adopted policies.

System dynamics discipline emerged in the late 1950s, as an attempt to address such dynamic, long-term policy issues, both in the public and corporate domain. Under the leadership of Jay W. Forrester, a group of researchers at M.I.T. initiated a new field then named Industrial Dynamics. (see "On the history, the present, and the future of system dynamics," EOLSS on-line, 2002). The first application area of the methodology was the strategic management of industrial problems. The main output of this research was the publication of *Industrial Dynamics*, the seminal book that introduced and illustrated the new methodology in the context of some classical industrial/business problems. The next major project was *Urban Dynamics*, presenting a dynamic theory of how the construction of housing and businesses determine the growth and stagnation in an urban area. With this application, "industrial dynamics" method moved to the larger domain of socio-economic problems and was eventually renamed "system dynamics." The second application in the larger socio-economic domain was *World Dynamics* (and *Limits to Growth*). These models show how population growth and economic development policies can interact to yield overshoot and collapse dynamics, when crowding and overindustrialization exceed the finite capacity of the environment. In a short period of time since the late 1970s, applications have expanded to a very wide spectrum, including national economic modeling, supply chains, project management, educational problems, energy systems, sustainable development, politics, psychology, medical sciences, health care, and many other areas. In 1983, the International System Dynamics Society was formed. The current membership of the Society is over 1,000. The number of worldwide practitioners is probably much higher than this number.

The purpose of the Theme articles (see Knowledge in depth) is to present a thorough exposition of the major aspects of system dynamics method, including its philosophical and historical roots, its technical components, selected exemplary applications, and specific illustrations of its potential in sustainability discourse. The Theme articles, written by experts in respective domains, are grouped under some natural "topics." We start with "System dynamics in action," EOLSS on-line, 2002, a small collection of exemplary applications to give the reader an idea of how the methodology is applied, illustrating the breadth, as well as the depth. The second topic consists of articles discussing the historical, conceptual and philosophical foundations of the field. The third topic of the Theme covers some fundamental concepts and tools of the system dynamics methodology. We next focus on selected technical/mathematical issues in modeling, numerical problems in simulation and other software considerations. The fifth topic is about policy improvement and implementation issues, covering public policy as well as business strategy, plus some general concepts and procedures of implementation. Finally, the last topic consists of six different applications of system dynamics approach and methodology in areas closely related to sustainable development.

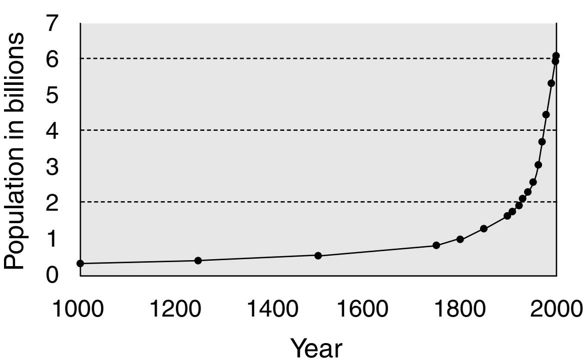
The goal of this particular Theme-level article is to provide a comprehensive overview of system dynamics methodology, including its conceptual/philosophical framework, as well as the technical aspects of model construction, analysis and policy design.

2. DYNAMIC PROBLEMS AND SYSTEMIC FEEDBACK PERSPECTIVE

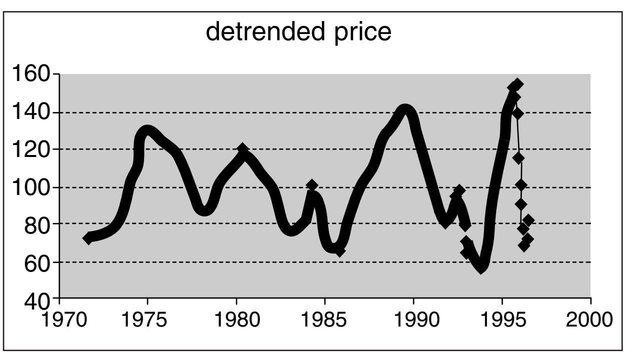
2.1. Dynamic feedback problems

The term "dynamic," in its general sense, means "in motion" or "changing over time." Dynamic problems are characterized by variables that undergo significant changes in time. Inventory and production managers must deal with inventories and orders that fluctuate; city administrations face increasing levels of solid waste, air and water pollution; wildlife managers are concerned about declining species diversity worldwide; citizens raise their voices against escalating arms race; at a personal level, we are concerned when our blood pressure or our heartbeat is unstable, or when our temperature goes up; national leaders are worried about increasing unemployment and inflation levels; the small company is in danger when a few years of fast growth is followed by a sharp decline in the market share. In each one of these cases, there are one or more patterns of dynamic behavior that must be managed (controlled, altered or even reversed). Figure 1 illustrates some typical dynamic patterns observed in real life: explosively growing world population, oscillating commodity (pulp) prices, exponentially declining cocaine prices, and growth-then-collapse in revenues (of Atari, Inc). Yet the defining property of a dynamic problem in our sense (in *systemic, feedback* sense) is not merely the variables being dynamic. More critically, in a systemic dynamic problem, the dynamics of the variables must be closely associated with the operation of the *internal structure* of some identifiable system. It is said that the dynamics is essentially caused by the internal feedback structure of the system. (*endogenous* perspective). Thus, oscillating inventories is a systemic, feedback problem because oscillations are typically generated by the interaction of ordering and production policies of managers. It would not be a systemic problem, if the oscillations were determined by an external force like fluctuating weather conditions: although it would still be a dynamic problem in its technical sense. In this latter case, there is not much the inventory managers can do to eliminate or reduce the oscillations. If, as allowed by a new deregulation, a multinational chain colonizes the grocery market, which in turn causes an unavoidable decline in small grocery store sales, this would not be a systemic feedback problem. There is not much "management" the small grocery store owner can do in order to reverse the deregulation or influence the policies of the multinational chain. The importance of the distinction is that if/when the dynamics are dictated by forces external to the system; there is not much space or possibility for managerial control and improvement. Systemic feedback problems on the other hand, necessitate dynamic, continuous managerial action.

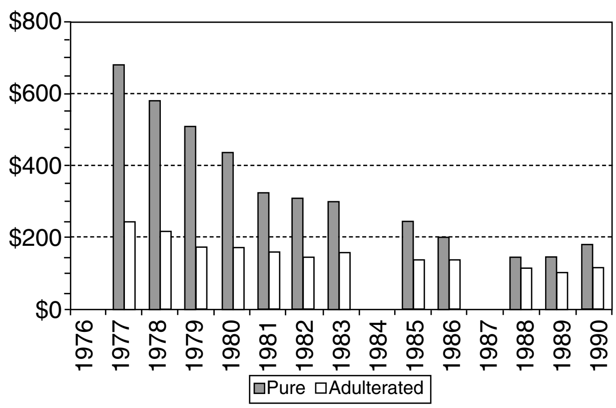
Dynamic management problems in real life are typically feedback problems: we take managerial actions, observe the results, evaluate them and take new actions, yielding new results, observations, further actions, and so on, which constitutes a "feedback loop." Feedback loops exist not only between the managerial action and the system, but also in between the various elements within the system. That is, most dynamic management problems are also "systemic" in nature. The main purpose of system dynamics methodology is to understand the causes of undesirable dynamics and design new policies to ameliorate/eliminate them. Managerial understanding, action and control are at the heart of the method. System dynamics thus focuses on dynamic problems of systemic, feedback nature.



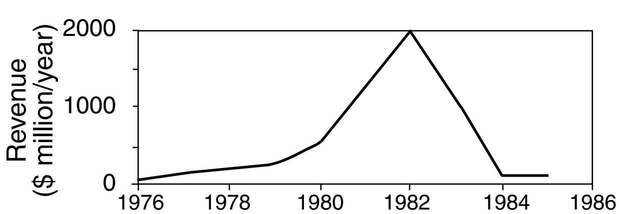
(a) World population growth (See "The ECOCOSM Paradox," EOLSS on-line, 2002).



(b) USA pulp prices (Deflated by CPI and all trends removed)



(c) Retail price per gram of cocaine (in 1990 dollars) (See "A Dynamic Model of Cocaine Prevalence," EOLSS on-line, 2002).



(d) Boom and bust: sales and operating income of Atari, Inc. (See "Market growth, collapse and failure to learn from interactive simulation games," EOLSS on-line, 2002).

Figure 1. Illustrations of some typical dynamic patterns observed in real data
(a) Exp. growth, (b) Oscillation, (c) decay, and (d) Boom-then-collapse.

2.2. Systems, problems, and models

The term *system* refers to "reality" or some aspects of reality. A system may be defined as a "collection of interrelated elements, forming a meaningful whole." So, it is common to talk about a financial system, a social system, a political system, a production system, a distribution system, an educational system, or a biological system. Each of these systems consists of many elements interacting in a meaningful way, so that the system can presumably serve its "purpose." But it is not trivial for a system to serve its purpose effectively: the global socio-economic system is still facing millions literally starving to death – certainly not an intended result; on the other hand our economic development generates solid waste, air and water pollution levels that threaten the sustainability of life on earth; commodity and financial systems alike generate highly unstable fluctuations worldwide; national economies are often unable to control the simultaneous problems of budget deficit, inflation, and unemployment; small companies typically enjoy a fast growth initially, only to be followed by a sudden collapse – an inevitable result of the unbalanced growth itself; as individuals, we must often deal with health problems such as high blood pressure or cholesterol that our very life style creates, and so on. In short, systems at all levels, scale and scope, while solving one set of problems, simultaneously "produce" other complex challenges.

A common scientific tool used in investigating problems and solutions is *modeling*. A *model* can be defined as "a representation of selected aspects of a real system with respect to some specific problem(s)." Thus, we do not build "models of systems," but build models of selected aspects of systems to study specific problems. The crucial motivation, purpose that triggers modeling is a problem. The problem can be practical (such as increasing levels of unemployment, declining species population, or collapsing stock prices) or theoretical (such as analyzing a cognitive theory of how knowledge is acquired, or testing the validity of Marx's theory of class struggle, or whether long-term ecological/environmental sustainability is possible in a capitalist system). In any case, without a problem-purpose, "modeling a system" is meaningless. One does not build a model of a national economy, or a species population, or stock market. One builds a model of some selected elements and relations in a national economy that are likely to have caused a recent upward trend in unemployment; or a model of selected factors believed to play a strong role in an unstoppable decline in the population of a species of concern; or a model that selectively focuses on those factors that are likely to explain a crash in a specific stock market in a specific time period. In the theoretical domain, the principle is the same: one builds a model to study a specific problem/theory so as to contribute to the debate in the relevant community.

Models can be of many types: *Physical* models consist of physical objects (such as scaled models of airplanes, submarines, architectural models, models of molecules). *Symbolic* models consist of abstract symbols (such as verbal descriptions, diagrams, graphs, mathematical equations). System dynamics models are symbolic models consisting of a combination of diagrams, graphs and equations. In another dimension, models can be *static*, representing static balances between variables, assumed to be constant in a time period (such as an architectural model or a mathematical equation representing the relation between price, supply and demand at a point in time). Or they can be *dynamic*, representing how the variables change over time (such as an aircraft simulator to train pilots, Newton's laws of motion, or a mathematical model of price fluctuations in commodity markets). Another typical classification of models is: *descriptive* versus *prescriptive*. Descriptive models describe how variables interact and how the problems are generated "as is," they do not state how the system "should" function in order to eliminate the problems. Prescriptive (often optimization) models however assume certain "objective functions" and seek to derive the "optimum" decisions that maximize the assumed objective functions. Nonlinear dynamic feedback problems are typically mathematically impossible to be represented and solved by optimization models. System dynamics models are thus descriptive models. The policy recommendations are derived not by the model, but by the modeler, as a result of a series of simulation experiments. Finally, dynamic models can be *continuous* or *discrete* in time. In time-continuous models, change can occur at any instant in time (such as air temperature, humidity, or population of a city), whereas in time-discrete models, change can only occur at pre-defined discrete points in time (such as salaries changing in multiples of months, or student grade point average changing each semester). Real dynamic systems consist of both types of dynamics. So a system dynamics model can be continuous, discrete or even hybrid. In practice, if the discrete time steps associated with the discrete variables is small enough compared to the time horizon of interest (such as a model involving salary dynamics, simulated for decades) one can safely do the time-continuity assumption. The rationale behind this approximation is that continuous-discrete hybrid models can be very cumbersome to build, analyze, and communicate. Continuous system dynamics models are mathematically equivalent to *differential* (or *integral*) equations, whereas discrete models are *difference* equations. In sum, typical system dynamics models are descriptive, continuous or discrete dynamics models, focusing on policy problems involving feedback structures.

2.3. Structure and behavior

The *structure* of a system can be defined as "the totality of the relationships that exist between system variables." In a production system, the structure would include the material and information flows related to production, where and how the various stocks are stored and shipped, how the ordering and production decisions are made, and so on. The structure of the system operates over time so as to produce the *dynamic behavior patterns* of the system variables. It is said, "the structure creates the behavior." The structure of the production department operates over time and generates production rate dynamics, raw material orders, oscillating inventories, and depending on system boundary, dynamics of product quality and sales. Dynamic behaviors of a few variables observed in different real systems were already illustrated in Figure 1. Figure 2 presents a generic template of most dynamic behavior patterns typically encountered in dynamic feedback problems. As seen in the template, the common dynamic behavior categories are: constant, growth, decline, growth-then-decline, decline-then-growth, and oscillatory. In each category there are different variants of the representative category, as seen in the figure. Furthermore, in reality these basic dynamic patterns can combine in various ways, such as oscillations followed by collapse or growth followed by decline-then-growth. Examples of such dynamic behavior patterns will be seen throughout this Theme.

The structure of a real system can naturally be extremely complicated, hence not exactly or completely known. But we do know that there is a structure that operates to produce the dynamics of system variables. The *structure* of the model is a representation of those aspects of the real structure that we believe (or hypothesize) to be important, with respect to the specific problems of concern. In the production example, the problem may be large oscillations in inventories or a persistent decline in product quality. These two different problem-orientations would yield different model structures. The structure of the model comprises only those parts of the real structure that play a primary role in creating the problematic behavior patterns. The structure of a production model to study the problem of oscillating inventories must include all variables, factors and relations responsible for inventory oscillations

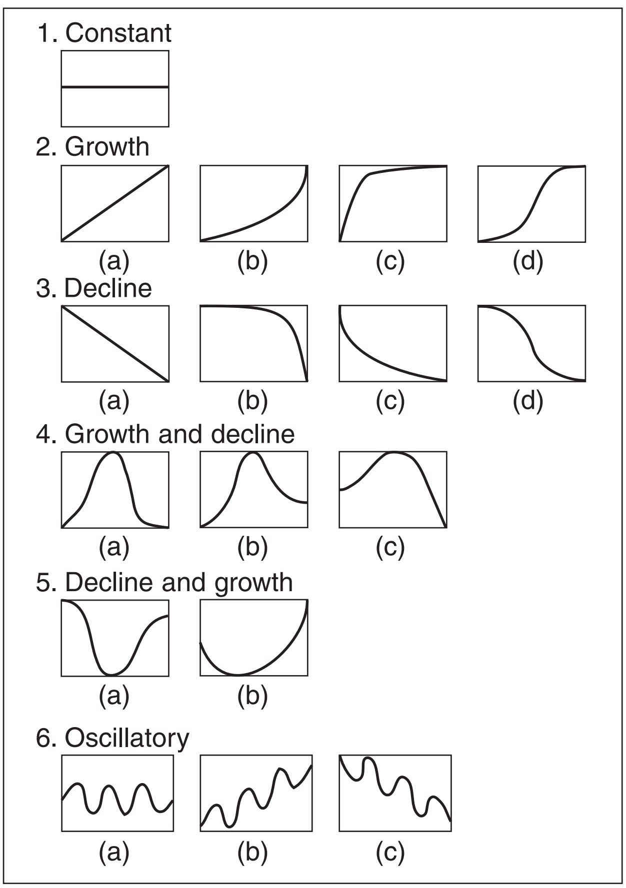


Figure 2. Categories of basic dynamic behavior patterns typically seen in systemic feedback problems

The structure of a system dynamics model consists of the set of relations between model variables, mathematically represented in the form of equations. That is, the structure of a system dynamics model is a set of differential and/or difference equations. The analytical (mathematical) solution of a dynamic model, if obtainable, would give the exact formula for the dynamic behaviors of variables. So, one way of obtaining the dynamic behavior of a model is solving it analytically. This is often possible in linear cases, but very rarely possible in non-linear ones. In such cases, the dynamic behavior of the model is obtained by *simulation*. Simulation is essentially a step-by-step operation of the model structure over compressed time. Much like the operation of the real structure over real time, the model structure operates over simulated time, so that the dynamics of model variables gradually unfold. A crucial notion in system dynamics method is that the term "model" refers strictly to the structure, to the set of equations describing it. The dynamic behavior of the model, whether represented in equations or plotted graphically, is not a "model" in system dynamics sense. The dynamic behavior is the output or the result of a model, very different conceptually from the model itself. In some disciplines such as forecasting, an equation representing a dynamic behavior would be legitimately called a model; but the structure-behavior distinction is crucial in system dynamics method.

Example 2.1

Consider a simple population dynamics of a single species on a large, isolated island. The structure of this system in real life would consist of males and females mating and reproducing, growing up, getting old, and eventually dying. Assuming that there is plenty of food, no competition, no epidemic, and a large island, if we start with a small number of animals (males and females), the operation of this structure over time would create an exponential growth in the population.

Assume that our purpose is to study this growth in population, its doubling time and its sensitivity to different birth and death rates. In the real system, there would be many details like some members being weak and unable to find food or mate, territorial fights, different categories of death (old age, fights, disease) geographical distribution of the population, weather conditions, seasonal changes in habitat, and so on. But for the simple purpose stated above, these details can perhaps be omitted and a simple model would be:

$\frac{dx}{dt}$ = *births* – *deaths*, which states that the rate of change of population is (*births* – *deaths*), where $x(t)$ denotes the population of the species at time t, and *births* and *deaths* must be specified.

The simplest formulation for *births* and *deaths* is the assumption that they are proportional to the population: *births=bx* and *deaths=fx,* where b is the birth fraction per year and f is the death fraction per year. Thus:

$$\frac{dx}{dt} = bx - fx$$

The above model is a differential equation. Alternatively, the same equation can be represented in integral form:

$$x(t) = x(0) + \int_0^t (bx - fx)\,dt,$$

which states that the population at time t is given by its value at time 0 plus the sum of all births minus all deaths between time 0 and time t. Population at time 0 is assumed to be known as $x(0)$.

In order to be able to solve this model, we must make one final assumption about the parameters b and f. In the simplest case, these fractions are assumed to be constant. Then, the simple differential equation model for population growth becomes:

$$\frac{dx}{dt} = (b - f)\,x = kx,$$

where we let $k = (b - f)$, also a constant.

To solve, use separation of variables and integrate:

$\int (\frac{1}{x})\ dx = \int k dt$, which yields:

$\ln x + C1 = kt + C2$, where $C1$ and $C2$ are arbitrary constants of integration. Combining:

$\ln x = kt + C3$, where $C3$ is another constant. Taking exponential of both sides:

$x = e^{kt+C^3} = e^{C^3} e^{kt} = Ce^{kt}$, where C is yet another constant. To evaluate it, use $x(0)$:

$x(0) = Ce^0$, or $C = x(0)$. Thus, the complete solution is:

$x(t) = x(0)\,e^{kt}$.

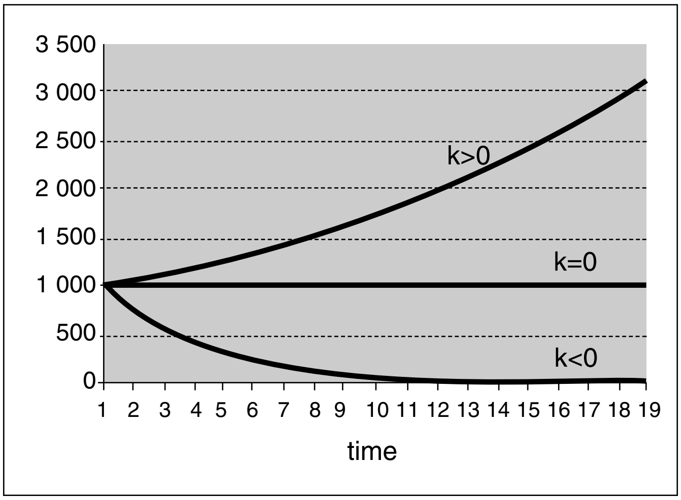
We can plot this solution for possible values of k and obtain the dynamic behaviors shown in Figure 3(a). Note that the dynamic behavior of the model depends on the value of $k = b - f$. If $b > f$, then the behavior is an exponential growth, if $b < f$, then the behavior is a negative exponential decline and if $b = f$, then the population stays constant.

The *structure* of this system is represented by the differential (or integral) equation given above. Its *dynamic behavior* is described by the solution equation ($x(t) = x(0)\,e^{kt}$). Alternatively, we typically represent the structure by a *stock-flow diagram* (and associated simulation equations). The dynamic behavior is then obtained by simulation and presented graphically. The stock-flow structure of the simple population model built in STELLA software and the simulated dynamic behavior are shown in Figure 3 (b) and (c).

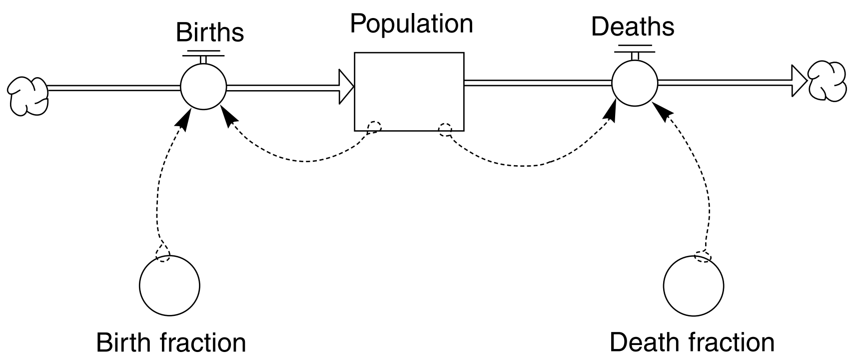
2.4. Principles of "systemic feedback" approach

System dynamics discipline deals with dynamic policy problems of systemic, feedback nature. As discussed above, such problems arise from the interactions between system variables and from the feedbacks between the managerial actions and the system's reactions. The purpose of a system dynamics study is to understand the causes of a dynamic problem, and then search for policies that alleviate/eliminate them. This specific purpose necessitates the adoption of a particular philosophy of modeling, analysis and design. This philosophy can be called "systemic feedback" philosophy (or approach or thinking or perspective). Systemic feedback thinking has important historical links with various disciplines and philosophies, namely General Systems Theory (Ludwig von Bertalanffy), Systems Theory and Sciences (i.e. Kenneth Boulding and Herbert Simon), Systems Approach (i.e. West Churchman), Cybernetics (Norbert Wiener) and Feedback Control Theory (Gordon Brown). (see "The role of system dynamics within the 'Systems' movement," EOLSS on-line, 2002) In a sense, systemic feedback philosophy integrates systems theory with cybernetics and feedback control theories. It is thus a unique systems theory, placing critical emphasis on the dynamic and feedback (closed-loop) nature of policy problems. From this perspective, it is possible to identify some principles that are absolutely essential to systemic feedback approach:

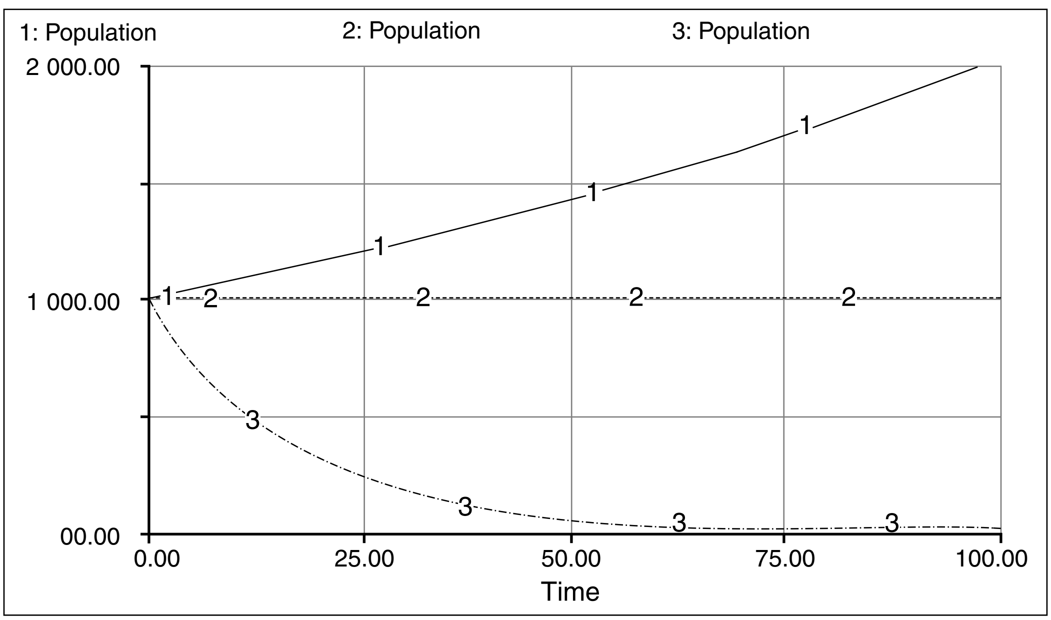
Principle 1: Importance of causal relations (as opposed to mere correlations).The purpose of a system dynamics study ("understanding and improving the dynamics") is very different from short-term prediction (forecasting) of future values of variables. The very purpose of system dynamics study requires that the model consist of causal relations, not mere statistical correlations. It is possible to generate excellent short-term forecasts by non-causal correlational models, but impossible to understand and control dynamic problems. For instance, swimsuit sales and ice cream sales are highly positively correlated (both going up in spring-summer season). If we have data on swimsuit sales (x), we can construct a correlational model $y = f(x)$ that would provide excellent forecasts for ice cream demand. For this narrow forecasting problem, the above model is very functional. On the other hand, assume that a company is faced with a consistent decline in ice cream sales after several years of growth and our objective is to understand (and reverse) this problematic dynamics. In this case, the correlational model $y = f(x)$ would be of no use, since this model gives no clue as to what "causes" the dynamics of ice cream sales. The input variable (swimsuit sales) has no causal influence on ice cream sales (y) whatsoever. Ice cream sales did not go down *because* swimsuit sales went down and they will certainly not improve by improving swimsuit sales (which is most likely not



(a)



(b)



(c)

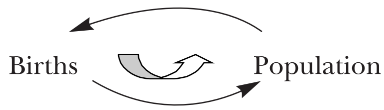
Figure 3. Illustration of model structure and dynamic behaviour (a) the solution of a simple population model, (b) its structure (stock-flow representation) and (c) its simulated dynamic behavior

even part of our business)! Similarly, one can show that rainfall and skin cancer incidences are negatively correlated. (In cloudy regions, rainfall goes up and skin cancer goes down due to reduced sunrays). But rainfall does *not cause* skin cancer incidence to go down. If we confuse this negative correlation with negative causation, we would start marketing bottled rainfall for people to rub on their skins three times a day!

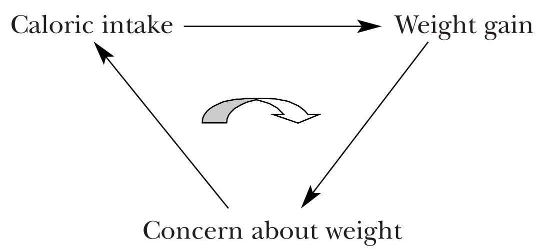
A causal relation $y=f(x)$ means that the input variable (x) has some *causal influence* on the output variable (y). Statements like "an increase in caloric intake causes weight gain" / "increased death rate causes the population to decline" / "increased pesticide application causes an increase in bird deaths" or "an increase in price causes the demand for a given commodity to go down" all reflect simple cause-effect relations. In each of these examples, if the cause variable is changed, one expects "some degree of change" in the effect variable. The term "expects" is important in the definition of "causality" in systemic feedback modeling. Whether the expected change actually occurs or what kind of change is observed (in the model or in reality) depends on many other factors. Each causal relation in a system dynamics model is a "*ceteris paribus*" argument: "other things being equal" the causality is expected to yield the stated effect. Since there are many other varying factors in a dynamic feedback model (and in reality), ceteris paribus condition would not/should not typically hold. Thus, an increase in price may not actually result in a decline in demand, because the price of the competitor product may have increased even more or the quality of the product may have simultaneously gone up. So the notation $X \rightarrow Y$ reads "other things being equal, a change in X causes a change in Y."

From a philosophical perspective, causality is a very difficult and debatable notion. There is no universally accepted, non-problematic definition of causation in philosophy. Yet the notion of causality needed for system dynamics modeling is an operational, practical one: non-controversial cause-effect relations, well established either by direct real-life experience or by scientific evidence in the literature. In adopting this definition, the main idea is to avoid using mere statistical correlations in lieu of causation. Once this is understood, in this operational, practical sense, there is normally no disagreement on what is "causal" and what is merely "correlational." (The correlational and causal examples given in the above paragraph should make the point clear. More examples will be seen later in section 3).

Principle 2: Importance of circular causality (feedback causation) over time. Identification of one-way causal relations described above is only the first step in dynamic feedback model conceptualization. The next crucial phase is the identification of dynamic, circular causalities (feedback loops) over time. One-way causality is in a sense "static" causality at a point (or small interval) in time. The relation births $\rightarrow$ population states that births are a cause and population is an effect. But when examined dynamically over time, it is also true that population causally affects births. (The more people, the more births there will be). Thus we have:



The above picture says that over time, births determine population and population determines births. Such a circular causality is only possible dynamically; it requires passage of time, because at a fixed instant in time it would be impossible for births to determine population *and* be determined by it. But over time, cause and effect (births and population) continuously trade places. More births mean higher population, which causes even more births, causing even higher population, and so on. The operation of this loop over time would create exponentially growing population. The feedback loop is in this sense the "engine" of change. (As will be seen later, interaction of multiple loops would constitute a much more sophisticated engine, creating much more sophisticated dynamics). In general, for a feedback loop to form between two variables, one or more variables would intervene. For instance, using the very first causality example above:

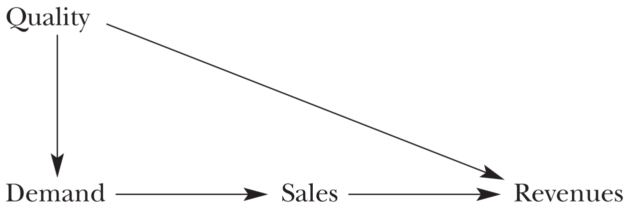


The above picture states that the more calories I take, the more weight I gain and as I realize my weight gain, I become concerned about it, which leads me to cut down on my caloric intake. In the original example, caloric intake was cause and weight gain was the effect. In this dynamic version, we observe that weight can be in time an indirect *cause* of caloric intake, via the new variable "concern about weight." Notice also that the dynamic behavior of this loop would be very different from the births-population loop; it would keep the caloric intake under control. The behavior of this loop would be a convergent one, rather than a divergent one. A more technical discussion of these two types of loops will be given later in the following section.

The above 3–variable loop illustrates another important sub-principle: A cause-effect arrow in a model must represent a "direct causality" between two variables, given the other variables in the model. In the above example, it would be wrong to draw a link from weight to caloric intake directly (caloric intake $\leftarrow$ weight), since the effect must really go through the intervening variable "concern about weight." As a more typical example, consider Quality of a product, Demand, Sales and Revenues. A simple causal diagram would be:

Quality $\longrightarrow$ Demand $\longrightarrow$ Sales $\longrightarrow$ Revenues

The diagram says: higher quality causes an increase in demand, meaning higher sales, hence higher Revenues, (assuming fixed price). In daily talk, one often summarizes the above diagram by "quality causes increased revenues." An erroneous causal diagram would be obtained, if one were to implement this simplification in the above diagram:



(a) Wrong inclusion of "indirect" causality.

Quality ⟶ Revenues

(b) "Correct" in the context of given variables.

The conceptualization in (a) is clearly wrong, as it is "double counting" the effect of quality on revenues. Quality has no known direct influence on Revenues, other than influencing it via Demand. Also note that the diagram would still be wrong, even if one eliminated the arrow from Quality to Demand (to avoid double counting). Given that the model already has the variables Demand and Sales, it would be illogical for the Quality to affect the Revenues directly. However, the diagram in part (b) would be a correct simplification, given the assumption that Demand and Revenues are not needed in the model. In this case, the "indirect" causal arrow from Quality to Revenues is an acceptable simplification. Thus the principle of "direct causality" is important in system dynamics, but it is also relative: "direct causality in the context of other variables in the model."

Principle 3: Dynamic behavior pattern orientation (rather than event-orientation. It is crucial to reiterate that the purpose of a system dynamics study is to understand the causes of a dynamic problem, and search for policies that alleviate/eliminate them. Dynamic problems are characterized by undesirable performance patterns, not by one or more isolated "events." In ordinary daily life, we all react to events: a sudden drop in the stock prices, a jump in the interest rates, resignation of a cabinet, or the September 11 attack on the World Trade Center are dramatic examples. Dynamic feedback approach argues that such events can not be analyzed or understood in isolation from their past dynamics. When isolated by their past histories, and by the dynamics of related variables, events seem random, unavoidable, and externally caused. But dynamic systems approach argues that most important events occur as a result of some accumulations (often hidden) reaching threshold levels over time. Short-term event orientation makes it impossible to understand the real historical and structural causes of events. Hence, analysis and understanding requires that we move away from the common event-orientation and focus on historical behavior patterns associated with potential events. With the proper behavior-pattern orientation, the goal is to construct a hypothesis (a model structure) that explains why and how the dynamic patterns of concern are generated. (Typical dynamic behavior patterns were illustrated in Figures 1 and 2).

Principle 4: Internal structure as the main cause of dynamic behavior (Endogenous perspective). The structure of a system was already defined as "the totality of the relationships that exist between system variables." One way of representing the structure of a system is diagramming the causal links and (especially) loops that exist between the variables. As stated above, the interaction of the feedback loops in a system is the main engine of change for the system. That is why it is said: "the structure causes the behavior of the system." This principle is critical in a system dynamics study, because the purpose is to understand the causes of an undesirable behavior and try to improve it. The principle can be better illustrated by an example.

Example 2.2

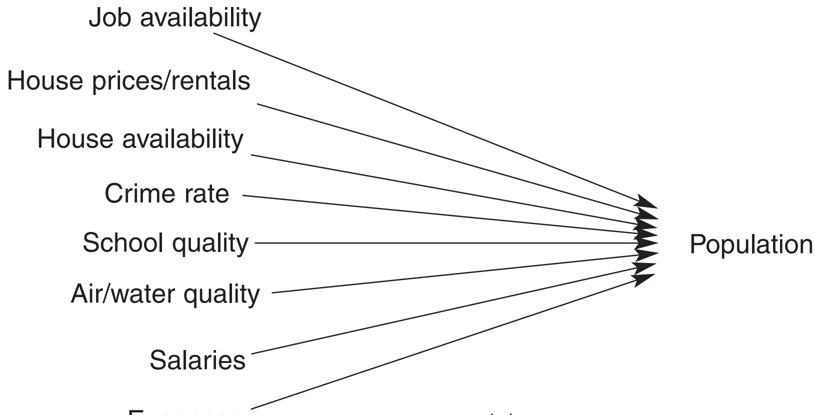
Consider the dynamics of the population of a modern city – perhaps a strong early growth followed by stagnation. A static and exogenous model for population would be:

Population = f(Job availability, Salaries, Expenses, House prices, House availability, Crime rate, School quality, Air/water quality. . .)

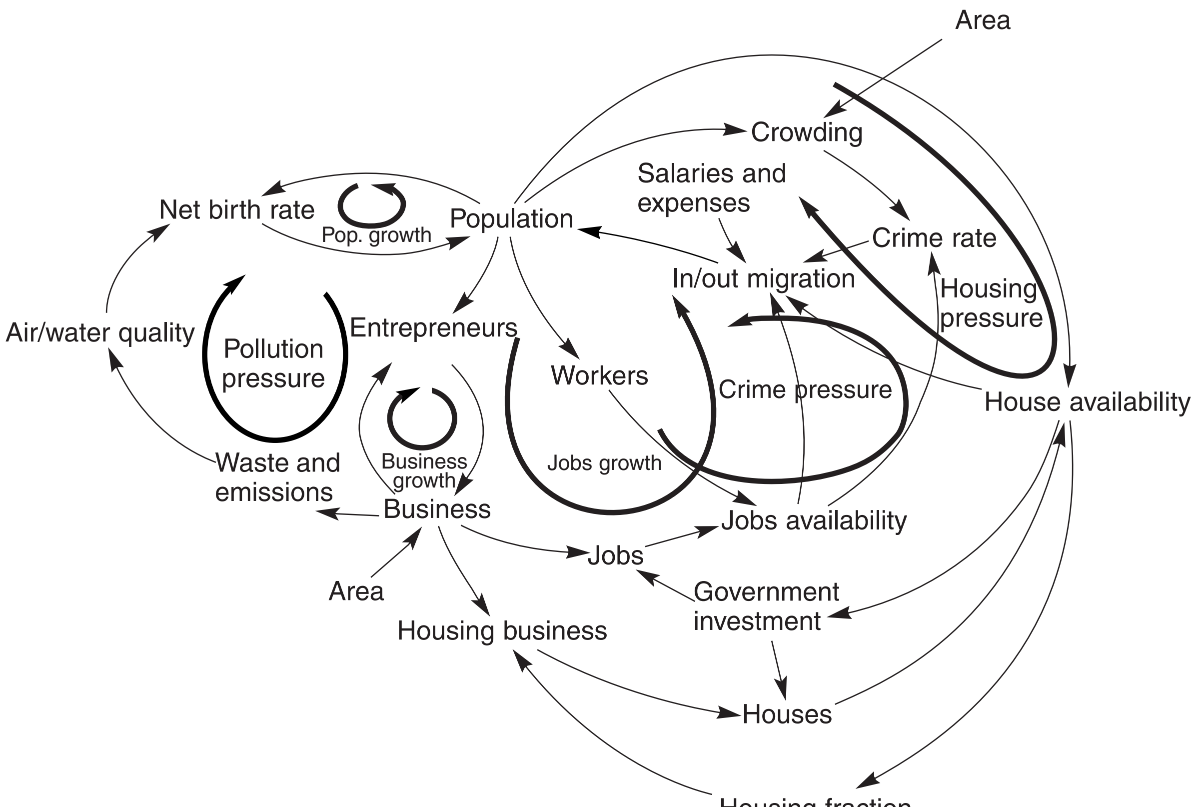
A pictorial representation of this model is shown in Figure 4(a). This model basically says that it is possible to determine population level, once the values of the input variables (eight of them) are given.

This model has some very strong assumptions. It assumes that the input variables are strictly *independent* variables. That is, they are not influenced at all by Population, or by any of the other seven input variables. (These external inputs are called "forcing" functions in engineering and their existence makes the system "forced" or "non-autonomous"). The model in Figure 4 (a) is of course an extremely unrealistic situation. In reality, Population would strongly affect some of these "independent" variables and some of them would strongly influence others as well. Such a model therefore is not a causal model of population dynamics, and as such cannot give any information on why the population behaves the way it does. On the other hand, this model could produce very reliable predictions of the population levels, for given values of input variables. Thus, such an exogenous model can be useful for short-term population forecasting, but cannot be used at all for policy analysis and design purposes.

Contrast the above model with the one shown in Figure 4(b). This second one is a dynamic and endogenous account of the dynamics of city population. Note that in this diagram there is no obvious distinction of "input" and "output." All major factors in the city influence one another and ultimately their own dynamics, around many feedback loops. This example is designed deliberately such that most variables in the



(a)



(b)

Figure 4. Comparing (a) an exogenous, static model of city population with (b) an endogenous, dynamic model

two models are common, so that the two can be easily compared. For instance, Job availability in the second model is given by the ratio Jobs/Workers, not an input variable. Jobs depend on Businesses and Workers are a fraction of the population. Businesses are initiated by entrepreneurs, which depend on Population as well as Businesses. (A fraction of Population yields potential entrepreneurs, but how active they actually are depends also on existing businesses). Finally, Job availability influences the in/out migration, which in turn creates an increase or a decrease in Population. Thus, a feedback loop emerges: Population → Entrepreneurs → Businesses → Jobs → In/out migration → Population. This loop says that more people, more entrepreneurs, more businesses, more jobs, a more in-migration. This is probably one of the major loops that create typical early booms in newly industrialized cities. But this growth is not without opposition; there are other loops. For instance, more Businesses, more Waste/Emissions, lower Air/water quality and lower net birth rate. Thus, Population → Entrepreneurs → Businesses → Waste/Emissions → Air/Water Quality → Net birth rate → Population loop suppresses the indefinite population growth suggested by the previous loop. In a similar fashion, one can trace several other growth loops and balancing loops in the diagram. In the end, the dynamics of the population is determined as a result of complex interactions between these loops. The dynamics could be growth, collapse, growth-and-stagnation, growth-and-collapse, and so on. A technical discussion would require building of a complete model with equations, beyond the scope of this section. (For a discussion of the structure of a classical urban dynamics model and the resulting growth-stagnation dynamics, see "Urban dynamics," EOLSS on-line, 2002). The point of this example is to illustrate the principle of "internal structure as main cause of dynamic behavior." The example demonstrates that it is the internal structure (the nature and interaction of the loops, their relative strengths, and so on) that determines the dynamic behavior. This model is an endogenous theory about the dynamics of city population, because the dynamics is not merely imposed by external forces, it is internally determined. On such a model, it is possible to study the causes of, say growth-then-collapse of the population of a modern city and explore alternative policies. By contrast, the previous static and exogenous model is not a theory about population dynamics; it can be used for forecasting purposes only.

Principle 5: Systems perspective. For the system dynamics methodology to be applicable to a problem, the dynamics of the variables must be closely associated with the operation of the internal structure of a system. But what if the dynamics in the real problem are dictated by external variables? There are two possibilities: Rarely, it may be indeed true that by its very nature, the real system may be too vulnerable to external influences. We already mentioned such an example: If a multinational grocery chain opens a super-store next to a small grocery store, which in turn causes an unavoidable decline in small store sales, this would not be a systemic feedback problem. There is not much "management" the small grocery store owner can do in order to influence the policies of the multinational chain, economies of scale, regulations, and so on. Such extreme cases are simply "ill-chosen" system dynamics problems and there is not much the methodology can do. The problem itself is "too open" by definition. But more often, the argument that the dynamics of the system are caused by external forces is a result of narrow system conceptualization, not a property of the real problem. For instance, in the population example above, the first model was static and exogenous whereas the second one was dynamic and endogenous. But assume now that, lacking systems perspective, the second model omitted Businesses and related variables. In this case, Jobs availability, Water/air quality and Housing availability would all become external input variables. The dynamics of Population would then be determined essentially by these external input variables in the second model as well. In general, the modeler faces the challenge of adopting proper systemic perspective such that the major forces and interactions are included in the internal structure. This is the critical "model boundary" determination issue in system dynamics method. From systems perspective, the model boundary must be wide enough so as to have an internal structure rich enough to provide an endogenous account of the dynamics of concern.

The importance of the endogenous/exogenous distinction is that, if external forces dictate the dynamics, there is not much possibility for managerial control and improvement. In real life, perhaps as a self-defense mechanism, we tend to give much more weight to external forces compared to the internal factors. At a personal level, when I am behind in my research, I put all the blame on the university bureaucracy, too much unnecessary committee work, inefficient assistants, and even bad luck, all of which means that I have no fault. The failing manager blames the unfavorable macro economic conditions, vicious/unethical competitors, unprofessional suppliers and yes, bad luck too. The prime minister blames the opposition parties, the chamber of commerce, worker unions, the president (if from another party), hostile foreign governments, extraordinary weather conditions, and so on. "The enemy is out there." The major problem with this attitude is that it frees the management of the system (whether private life or a professional organization) from responsibility, which means no critical self-evaluation and no prospect for improvement. One of the challenges for a system dynamics study is to convince people that this type of defensive attitude is unproductive. From a systemic perspective, the dynamic problem is caused neither by the "external" enemy nor by the manager. There is no single person or entity to blame. The cause of the dynamic problem lies in a system structure that cannot cope with unfavorable external conditions. (The famous mass-spring system oscillates, when pushed by an external force. It does not oscillate *because* it is pushed. It oscillates because it has a spring, a structure ready to oscillate when touched). The cause is not the external enemy, but the way our system relates to/deals with the "external enemy." The internal structure of the model must include not only the relationships between the internal elements of a system, but also how the system relates/reacts to its environment, to the external forces. That is why "model boundary" determination is a subtle and critical task, quite dependent on the perspective and skills of the modeler. The model boundary is not automatically dictated by some natural "system boundary;" the modeler determines it.

2.5. Complexity of dynamic systems and necessity of modeling and simulation

It is often said that dynamic systems are "complex." This is especially true for non-linear feedback systems involving human actors – typical subject matter of system dynamics studies. Here are some of the main reasons why such problems are complex:

Dynamics

Dynamic problems are naturally harder than static problems. Variables change over time as they interact. The changes are not straightforward to predict. There are time delays involved between causes and effects and between actions and reactions. Dynamics of systems may be hard to predict by intuition even with only a few variables.

Feedback

The problem is further complicated when dynamics are created by operation of feedback loops. It means that which way the system will move is not easily predictable; the evolution path unfolds gradually and continuously determines its own path into the future. (Path-dependent dynamics). Feedback dynamics are harder to predict by intuition, because they require mental simulation of interactions of several loops simultaneously.

Non-linearity

Most system dynamics problems are non-linear. This means that the cause-effect relations between variables are not proportional. Non-linear effects are subtle, because a certain effect observed in one range may not be valid at all in another range. Non-linearity furthermore often means that there are "interaction effects" between variables. That is, doubling the advertising may increase the demand by 20 percent when the price is around US$100, but the same effect may be only 5 percent when the price is around US$125. Non-linearity is very hard to analyze not only intuitively, but also mathematically, especially when embedded in a dynamic feedback context.

Scale

As the number of variables increases, the complexity of the problem increases nonlinearly. With only three or four variables, even a non-linear feedback problem can be analyzed in most cases mathematically and perhaps intuitively. But even "small size" policy problems involve tens of variables. At this scale, a non-linear feedback problem immediately becomes impossibly hard to track – mathematically and intuitively.

Human dimension

Typical system dynamics problems involve human actors. So we must model not only the physics of the system (including information flows), but also how people react to situations, make decisions, set goals, make plans, and so on. This "human dimension" adds yet another layer of complexity. Human elements are much harder to model than the mechanical/physical aspects. There are no established, tested laws of how people behave, react, or make decisions. Quite often, the modeler must create his/her own theory of how the human actors would behave in the specific context of a given task and environment. Human dimension makes the study harder not only in modeling, but also in testing and analysis phases.

Cause and effect separated in time and space

The result of the above facts is that in a non-linear dynamic feedback model with several variables, the cause-effect relations become detached in time and space. When an action is applied at point A in the model with an expected immediate result at point B, this result may never be obtained and furthermore, some unintended effect may be observed at a distant point C, after some significant time delay. We make a change in our marketing activity to boost sales, but nothing substantial happens at first and then, after many months, we may observe some unintended consequences in our production department.

Intuitive inadequacy

We human beings are not naturally equipped to deal with this type of detached cause-effect relation. A baby touches the stove with his/her index finger, and the index finger burns, and it burns now. The child learns immediately that touching the stove burns the hand; the causality is easy to extract, because the cause and effect are close in time and space. Like all animals, we can immediately learn this type of cause-effect relation. Our very survival depends on this intuitive skill. Now assume a strange hypothetical world where when a baby touches the stove, nothing happens to his hand, but weeks later his nose starts bleeding. It would be close to impossible for the baby to be able to link the effect to its cause in this strange world, because cause and effect are detached in time and space. What is worse, the baby would probably link the effect to some wrong causes, as he surely did many other things involving his nose just a few days, hours, or minutes before the bleeding. Organizations, socio-economic systems, are unfortunately like this strange world. A new decision in one sector of the economy can have several unexpected effects in other sectors, after many months or even years. With our time and space-constrained intuition of causality, we are prone to make wrong causality inferences about effectiveness of critical managerial, public or personal decisions. Finally, our intuitive ability is further impeded by delays, errors, omissions, and bias in data/information that we observe in real life.

All of the above complexities lead to the following conclusion: Large scale, non-linear, dynamic feedback models are too complex to be even partially analyzed and understood by our natural intuition. So we need help. The help that we obtain is two-fold: first, *formal modeling*. We build a formal model in order to make our *mental model* explicit, rigorously analyzable and testable, making scientific improvement possible. Our mental models have some major drawbacks: they are vague, implicit, often biased, ambiguous, and non-testable. (see "Mental models of dynamic systems," EOLSS on-line, 2002). A formal model, in contrast is explicit, precise, less biased, unambiguous and testable. So, a well-constructed formal model can eliminate most of the weaknesses of mental models. But this is only half of the story. The other major issue is *analysis* of the formal model. The exact and most general scientific method is, of course, mathematical analysis. But the required mathematics is unfortunately almost always impossible for large, non-linear dynamic feedback models. Thus, the second major assistance we seek is *computer simulation.* Simulation is an *experimental* way of analyzing the problem, which is another standard method of analysis in science. But in simulation, instead of experimenting with the real system, we experiment with a model of the real problem. In simulation, the model structure operates over simulated time, just like the operation of the real structure over real time, so that the dynamics of the variables gradually unfold. Thus, a set of carefully designed experiments can be carried on the simulation model, yielding some results about the dynamic properties of the system, the causes of the problematic dynamics, and how they can be improved. The validity of the model is of course vitally important in this approach. In system dynamics studies, simulation experimentation is often the only feasible scientific method of analysis. The other two classical methods, mathematical analysis and experimenting in the real system are unfortunately too often impossible. The impossibility of real system experimentation comes from huge risks, costs, time delays and other practical impossibilities involved in experimenting with socio-economic systems. Simulation combines the cost and risk advantages of mathematical modeling/analysis, with the mathematically unconstrained power and flexibility of experimental analysis.

3. MODELING METHODOLOGY AND TOOLS

3.1. Steps of the system dynamics method

A typical system dynamics study goes through some standard steps. Although there will be variations depending on the nature of the problem and style of the modeler, main steps can be nevertheless summarized as follows.

Problem identification and definition (purpose)

A system dynamics project is done to study a dynamic problem (applied or theoretical). Selection and articulation of a meaningful dynamic feedback problem is critical for the success of the project. The problem must be not only dynamic, but also of feedback nature. Externally driven dynamic problems are not meaningful system dynamics topics. Dynamic problems are characterized by behavior patterns that may be observed in plotted data or they may be deduced from available qualitative information. In a theoretical study, the dynamics would be a hypothetical plot. Some of the sub-steps of problem identification are:

- Plot all the available dynamic data and examine the dynamic behaviors.
- Determine the time horizon (into the future and into the past) and basic time unit of the problem.
- Determine the *reference* dynamic behavior: What are the basic patterns of key variables? What is suggested by data and what is hypothesized if there is no data? What is expected in the future?
- Write down a specific, precise statement of what the dynamic problem is and how the study is expected to contribute to the analysis and solution of the problem. Keep in mind that this purpose statement will guide all the other steps that will follow.

Dynamic hypothesis and model conceptualization

The objective of this step is to develop a hypothesis, a theory that explains the causes behind the problematic dynamics. This must of course be an "endogenous" explanation. Later, this hypothesis will be converted to a formal simulation model and the validity of the hypothesis will be tested. Dynamic hypothesis can also be called a conceptual model; a model that describes our hypothesis, not yet in a formal, testable form. This step involves the following main activities:

- Examine the real problem and/or the relevant theoretical information in the literature.
- List all variables playing a potential role in the creation of the dynamics of concern.
- Identify the major causal effects and feedback loops between these variables.
- Construct an initial causal loop diagram and explore alternative hypotheses.
- Add and drop variables as necessary and fine-tune the causal loop diagram.
- Identify the main *stock* and *flow* variables.
- Use other conceptualization tools (like sector or policy diagrams) as suitable.
- Finalize a dynamic hypothesis as a concrete basis for formal model construction.

Formal model construction

In this step, the formal simulation model is built. This involves the following sub-steps:

- Construct the stock-flow diagram; the structure of the model
- Write down mathematical formulations that describe cause-effect relations for all variables
- Estimate the numerical values of parameters and initial values of stocks
- Test the consistency of the model internally and against the dynamic hypothesis (*verification*).

Model credibility (validity) testing

Is the model an adequate representation of the real problem with respect to the study purpose? Model credibility has two aspects: first, Structural: is the structure of the model a meaningful description of the real relations that exist in the problem of interest? And second, Behavioral: are the dynamic patterns generated by the model close enough to the real dynamic patterns of interest? (see "Model testing and validity," EOLSS on-line, 2002). Examples of structural tests are: having experts evaluate the model structures, dimensional consistency with realistic parameter definitions, and robustness of each equation under extreme conditions and extreme condition simulations. Behavior tests are designed to compare the major *pattern components* in the model behavior with the pattern components in the real behavior. Such pattern measures include slopes, maxima and minima, periods and amplitudes of oscillations (autocorrelation functions or spectral densities), inflection points, and so on. Two principles are critical: first, unless structural validity is established to begin with, behavior validity is meaningless in system dynamics, and second, behavior testing does not involve point-by-point comparison of model behavior with real behavior; it involves comparing of *patterns* involved in the two.

Analysis of the model

The purpose in this step is to understand the important dynamic properties of the model. This can be done very rarely (sometimes partially) by mathematical/analytical methods. Although it is impossible to find the solution equations of system dynamics models mathematically, we can sometimes find the constant equilibrium levels and determine their stability. (see "Equilibrium and stability analysis," EOLSS on-line, 2002). More typically, analysis is done by simulation experiments. A series of logically related simulation runs can provide quite reliable (although not exact) information about the properties of the model. These simulation runs are also called sensitivity tests, as they try to assess how much the output behavior changes as a result of changes in selected parameters, inputs, initial conditions, function shapes, or other structural changes. (see "Sensitivity analysis," EOLSS on-line, 2002).

Design improvement

Once the model is fully tested and its properties understood, the final step is to test alternative new *policies* to see to what extent they can improve the dynamics of the model. A policy is a decision rule, a general way of making decisions. Thus, a production policy would be represented by a set of equations that describe how the productions decisions are being made in the company. (An outcome of these equations would be a production *decision*). In this last step, alternative policies are designed and then tested by simulation runs. Policy improvement is a complicated task: The recommended policy must be realistic, considering the environment in which it will be implemented; the policy must be *robust*, in the sense that it should work under different environmental conditions and scenarios; interactions (positive or negative) of policies must be considered; transition dynamics – the response of the system during the transition from old policy to the new one – must be explicitly analyzed by simulation experiments as well. (see "Intellectual roots and philosophy of system dynamics," EOLSS on-line, 2002).

Implementation

This step is applicable if the system dynamics study is an applied one. Needless to say, it is a vitally important step, since the ultimate success of an applied system dynamics project means a demonstrable and sustainable system improvement. A successful implementation in some sense depends so much on project specifics that it cannot be prescribed general rules or procedures. On the other hand, there are some general aspects of implementation success that system dynamics researchers have attacked. (see "Implementation issues," EOLSS on-line, 2002). There are some important new developments that seek to enhance implementation success, such as group model building (see "Group model building, EOLSS on-line, 2002) and interactive learning environments (see "Modeling for learning and interactive environments," EOLSS on-line, 2002).

3.2. Stock and flow variables

In system dynamics models, it is essential to distinguish between two types of variables: *Stocks* and *flows*.

Stocks

They represent results of accumulations over time. Their values are "levels" of the accumulations. They are also called "states" as they collectively represent the state of the system at time *t*. The standard symbolic shape for a stock is a rectangle.
Stock Examples:

- population
- cash balance
- inventory of goods
- weight
- anger level
- knowledge level
- temperature
- glucose in blood

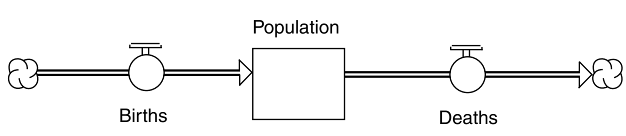
These examples show that stocks can be physical entities as well as information entities; they can be managerial control variables as well as natural/biological variables. Stocks are universal; they represent the critical accumulations on which the very existence of all sorts of life forms depend.

Flows

They directly flow in and out of the stocks, thus changing their values. They represent the "rate of change" of stocks. The symbol for a flow is an arrow (representing the direction of flow) and a valve (𝕋 or X representing the fact that the flow quantity is being regulated).
Flow examples:

- births, deaths
- income, expenses
- production, sales
- caloric intake, calories burned
- increase and decrease in anger level
- learning and forgetting (or obsolescence rate)
- heat in, heat out
- glucose intake, glucose consumption

Note that the flow examples above are naturally paired with the stock examples above. For instance, Births are an *inflow* and Deaths an *outflow* for the Population stock. The symbolic representation:



Mathematically, the above diagram states:

$$\frac{dPop}{dt} = births - deaths, \text{ in a continuous model.}$$

Or, to stress the accumulation nature of a stock:

$$Pop(t) = Pop(0) + \int_0^t (births - deaths)\,dt,\ Pop(0) \text{ given.}$$

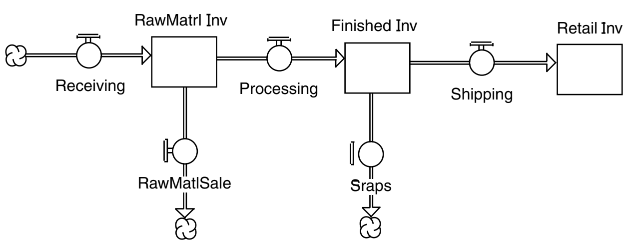
In numerical simulation, the same equation is approximately represented as:

$$Pop(t) = Pop(t - dt) + \ldots \text{ for } t = dt, 2dt, 3dt\ldots$$

Finally, if the model described a discrete dynamic system, then:

$$Pop(t) = Pop(t-1) + (births - deaths), \text{ for } t=1,2,3\ldots$$

In any case, we see that the standard stock equation is a basic conservation equation over time. To represent the conservation more explicitly, the flow Births would flow out of some stock (Babies in conception) and Deaths would flow in some stock (dead people). In the above diagram, "clouds" represent the implicit stocks "Babies in conception" and "Dead people." The cloud symbol means that these two stocks are *outside the model boundary*, so we do not need to track them. ("Cloud" usage is standard in "information" stock-flows such as "knowledge increase," because an increase in my knowledge does not decrease somebody else's knowledge: information flows typically are not conserved). More generally, in a stock-flow diagram there would be multiple stocks and flows, such as:



The above example says that there can be some sales directly from raw material inventory and some scraps from finished inventory. The computational stock equations in this case would be:

$RawMatrlInv(t) = RawMatrlInv(t - dt) + dt * (Receiving - Processing - RawMatlSale)$

$FinishedInv(t) = FinishedInv(t - dt) + dt * (Processing - Shipping - Sraps)$

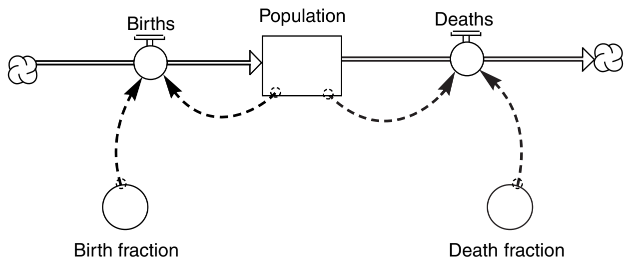
RetailInv(t) = RetailInv(t – dt) + dt * (Shipping)

In general, if a stock has j flows, the continuous stock equation is:

$$Stock(t) = Stock(0) + \int_0^t (\sum_j flows)\,dt$$

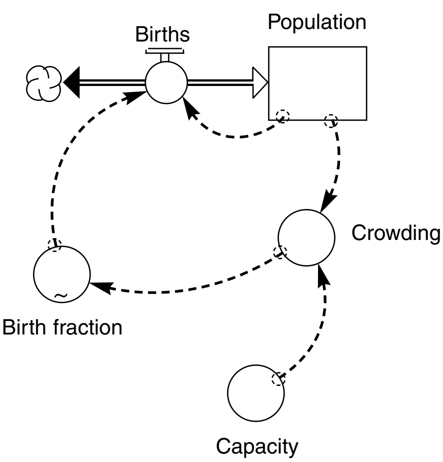
If a model has *n* stocks, it is said to be of *order n*. This is mathematically equivalent to a set of *n* 1st order differential equations, or a single *n*th order differential equation (that is, highest derivative involved is of order *n*).

The above examples do not represent complete models. To be solvable (analytically or by simulation), the equations for flows must be specified. For example, in the simple population model, the simplest specification for births and deaths would be: *Births = b*Population* and *Deaths = d*Population*, where *b*: constant birth fraction and *d*: constant death fraction. The stock-flow diagram would then be:



The "effect" arrows in this diagram show that *Births* = *f* (*Population, Birth Fraction*) and *Deaths* = *f* (*Population, Death Fraction*).

Finally, in writing equations for flows, one often needs to define some intermediate variables, called *auxiliary* or *converter* variables. For examples, consider a situation where *Birth Fraction* is not constant, but depends on the "crowding level" of the population, defined as *Crowding = Population/Capacity*. Birth Fraction would then be *f* (*Crowding*), where *f* (.) is typically a monotonically decreasing function. A stock-flow diagram including such an auxiliary variable (*Crowding*) is shown below:



In the above diagram, the flow is "net births" (*births – deaths*), represented as a "bi-flow" that flows in when its value is positive and flows out when it is negative. This formulation is obtained by defining the Birth Fraction function such that when *Crowding* < 1 it is positive (*births* > *deaths*) and when *Crowding* > 1 it is negative (*births* < *deaths*). (This model will be further studied below).

How to identify stocks and flows?

"Bathtub" is an excellent metaphor for stock variables. Water in the bathtub is the stock; water flowing in and flowing out are its flows. If a variable in the real problem fits this bathtub metaphor, it is a potential stock. Note that the stock examples listed above all fit the metaphor. A quick and effective test to identify the stocks is to "freeze" time and motion, and see which variables still persist. Since stocks are accumulated quantities, they are well defined, even if there is no time and motion. Population, inventory, cash balance, knowledge level. . . the stocks still persist; but their flows vanish: there can be no births, no deaths, no production, no sales, no expenditures, and no learning without passage of time. If a given stock were measured in certain units (say people, liras, items. . .) its flows would be in units/time period (people/year, liras/month, items/hour . . .). That is why flows become undefined without passage of time, but stocks persist.

The above test is important and presents a necessary condition for a variable to be a stock, but the condition is not sufficient. Not all variables that pass the test would be modeled as a stock; many would be instead modeled as *auxiliary* variables. Identifying a variable as a stock means deciding to model its flow variables. This would not only make the modeling more complicated, but addition of stock-flow structures would also complicate testing, analysis, and design phases, because stock-flow structures add to the dynamic and scale complexity of the model. Thus, the second important principle is that stock variables identified in a problem are believed to be *especially important accumulations* that decisions in the real system try to control or depend on. In the simple example above, although the variable Crowding passes the "freeze" test, we did not model it as an accumulation, because it was not believed to be important enough to justify the added complexity of formulating its flows in and out. Similarly, many such variables are approximated by auxiliary variables. Stock variables are those accumulations that seem most important with respect to the dynamic problem definition. Finally, potential stocks that vary too slowly compared to the time horizon of the model are modeled as constants. Selections of stocks are therefore also relative to the time horizon and time units of the problem. In some cases, even the stock-flow distinction can be relative. Cars traveling from Ankara to Istanbul can be a stock variable in one problem definition (short-term, micro dynamics), but it would be a flow variable (cars traveling per day) in a longer term, macro problem.

Importance of stocks

Stocks play a central role in dynamic feedback management problems for several reasons. Their control is often the primary responsibility of managers, since the survival of the system is often critically dependent on them: water in reservoirs, grain stocks, cash in banks, food stock in our kitchen, glucose in blood, and pollutant level in air. We not only control these variables, but also use their values as basis for action in managing other variables. Yet, controlling stocks is subtle and dynamically complex by their very nature:

- Stocks can be changed only via their flows: If we want to change the value of an inventory, we cannot directly and immediately change it. It can only be changed by changing its inflow and/or its outflow (production rate and/or sales). We can directly manipulate our caloric intake by changing our eating (flow), but cannot directly manipulate our weight. If changing our eating habit is hard, changing our weight will be indirect, delayed and much harder.
- Stocks have inertia: Since they have historically accumulated values, they cannot be easily changed. We can suddenly increase the inflow into the bathtub by 100 percent, but if it was already half-full, it will be a long time before the stock in the batch tub increases by just 10 percent. If I have a negative opinion of a political party, it would take quite an effort (and time) to change it.
- Multiple flows make the control even harder: Stocks and their flows may move in opposite directions. We may have increasing revenues, but the cash stock will still go down, if the expenses (outflow) are greater than revenues. We may have decreasing births *and* increasing deaths, but the population will still increase if births are nevertheless greater than deaths. In general, controlling the dynamics of a stock requires taking into account all of its flows simultaneously, which is not a trivial task, theoretically and practically.
- Stocks are the source of endogenous dynamics: Stocks make it possible for the inflows and outflows to differ. The differences between the flows accumulate in the stock. There are several water reservoirs around the city of Istanbul. Water in these reservoirs is a stock, streams feeding into the reservoirs are the major inflow and the water consumption of the city is the major outflow. Without reservoirs, the consumption of the city would have to equal the stream flows at all times, which means the consumption would fluctuate the same as stream flows do. Such wild fluctuations in water availability would make life in the city impossible. The reservoirs allow smooth water consumption (outflow), by buffering against seasonal fluctuations in the stream flows. The stocks influence their own flows in various ways. Since stocks are of such critical importance we continuously monitor their values and take actions to keep their values in some desired band. Stocks also cause time delays in systems. In the raw material supply chain diagram shown above, the fact that there are two stocks between the receiving of raw material and the shipments of goods to the retailers introduces a time delay between "Receiving" and "Shipping" flows. Similarly, information (or perception) delays also involve stocks. In sum, stocks play a central role in the creation of the endogenous dynamics in a system.

Basic stock-flow dynamics

Stocks *integrate* the flows, modifying and complicating the dynamics involved. Figures 5, 6 and 7 illustrate some basic flow-stock dynamics. Since stock integrates it flows, observe in Figure 5 that when the inflow and the outflow are constant, stock is linearly increasing or decreasing (or constant when inflow = outflow).

In Figure 6, we see the dynamics of the stock when the inflow and/or the outflow are changing linearly. Note that the stock is increasing or decreasing non-linearly (quadratic), due to integration.

Finally in Figure 7, the inflow is oscillating and the outflow is constant. The purpose of this example is to illustrate how integration creates time delay. Note that the stock imitates the inflow, except that it lags behind the inflow with phase lag of $\pi/2$, since integral of cosine is sine and sin (ß) = cos (ß-π/2). Also observe that amplitude of the oscillations in stock is more than the amplitude of the inflow (amplification).

3.3. Positive and negative causal effects and feedback loops

Positive and negative effects

We already mentioned that causal relation $x \rightarrow y$ means the input variable (x) has some *causal influence* on the output variable (y). A *positive* influence means: "a change in x, ceteris paribus, causes y to change in the same direction." Examples are:

Population ——>$^+$ Housing Demand
Births ——>$^+$ Population
Pesticide ——>$^+$ Bird deaths
Motivation ——>$^+$ Productivity

The "+" symbol denotes positive causality. (Some authors use the symbol "s" denoting "same direction"). When population goes up, housing demand goes up, ceteris paribus. (The ceteris paribus condition will be true in all causal relations, but will be omitted from now on to avoid repetition). Conversely, when population goes down, housing demand goes down. The same reasoning is true for the Pesticide → Bird deaths and Motivation → Productivity examples. But the second example is somewhat different: when births go up, population either goes up or it *decreases less than it would otherwise have been* (this would be true if, in spite of increased births, deaths were still higher). This situation is possible when the cause-effect relation is a flow-stock relation. Thus, the term "in the same direction" is subtle when flow-stock relation is involved. It means, "an increase (decrease) in x causes y to increase (decrease) above (below) what it would otherwise have been."

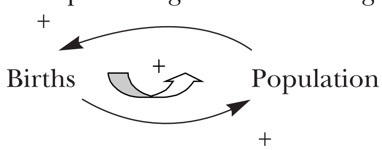
A *negative* influence means: "a change in x, ceteris paribus, causes y to change in the opposite direction." Examples are:

Unemployment ——>$^-$ Immigration
Deaths ——>$^-$ Population
Price ——>$^-$ Demand
Frustration ——>$^-$ Studying

The "-" symbol denotes negative causality. (Some authors prefer the symbol "o" denoting "opposite direction"). When unemployment in a country goes up, immigration goes down, ceteris paribus. Conversely, when unemployment goes down, immigration goes up. The same reasoning is true for the Price→ Demand and Frustration → Studying examples. But the second example, being a flow-stock relation, is again somewhat different: when deaths go up, population either goes down or it *increases less than it would otherwise have been* (this would be true if, in spite of increased deaths, births were still higher). Thus, the term "in the opposite direction" in this case means, "an increase (decrease) in x causes y to decrease (increase) below (above) what it would otherwise have been."

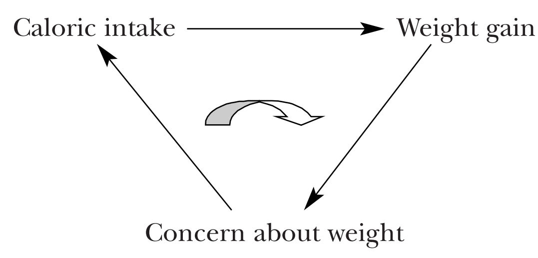
Positive and negative feedback loops

As seen before, a feedback loop is a succession of cause-effect relations that start and end with the same variable. It constitutes a circular causality, only meaningful dynamically, over time. The *sign* (or *polarity*) of a loop is the algebraic product of all signs around the loop. If the resulting sign is +, the loop is "positive" or "compounding" or "reinforcing."

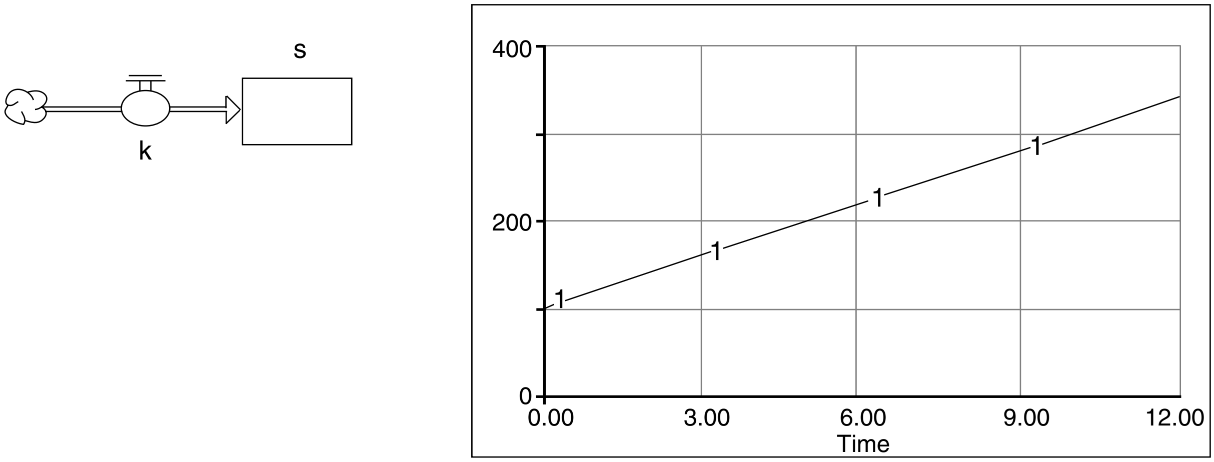


The above picture says that more births mean higher population, which causes even more births, causing even higher population, and so on. The operation of this loop over time would create exponentially growing population. This feedback loop reinforces or compounds an initial change. Not all positive loops create growth; some create *collapse* as will be seen later.

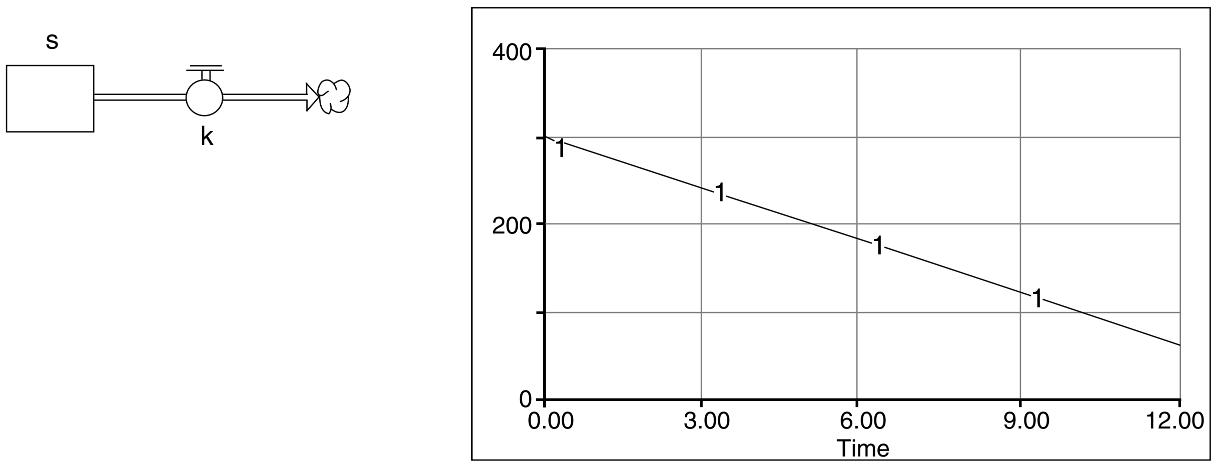
If the resulting sign of the loop is negative, then the loop is called "negative" or "balancing" or "goal-seeking." This type of feedback loop seeks a balance or a goal.



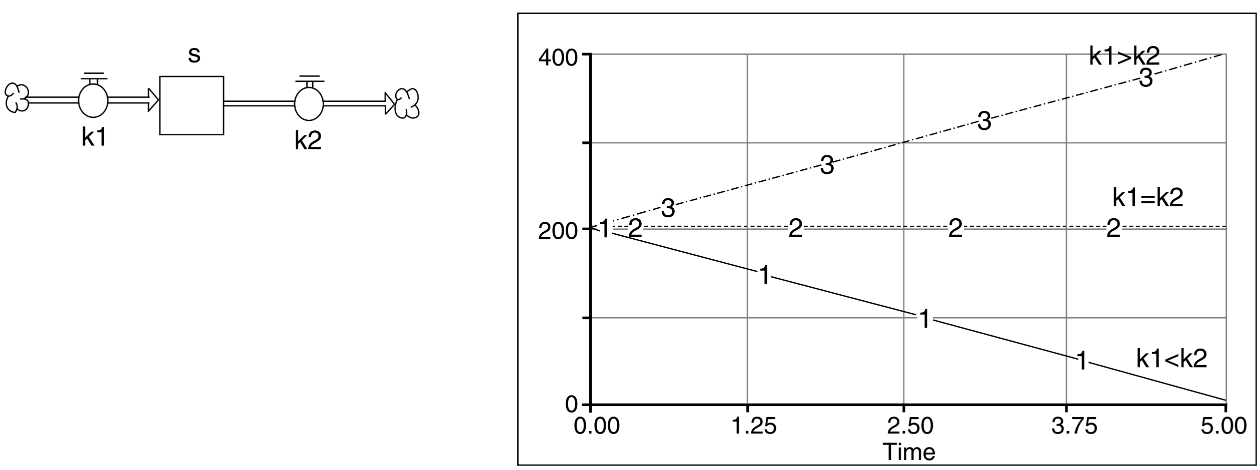
The above picture states that the more calories I take, the more weight I gain and as I realize my weight gain, I become concerned about it, which leads me to cut down on my caloric intake. This loop tries to keep the caloric intake (hence the weight) under control. The behavior of this loop would be a convergent one, rather than a divergent one. In this example, the person tries to control her weight around some "desired" ("goal") weight. In some other negative loops, there is no explicit or deliberate goal. For instance, consider a tank of water emptying itself out of a hole punched in the bottom. The outflow would



(a)



(b)



(c)

Figure 5. Dynamics of a stock: (a) with constant inflow only, (b) with constant outflow only and (c) with constant inflow (k1) and constant outflow (k2), when k1>k2, k1=k2 and k1<k2 respectively

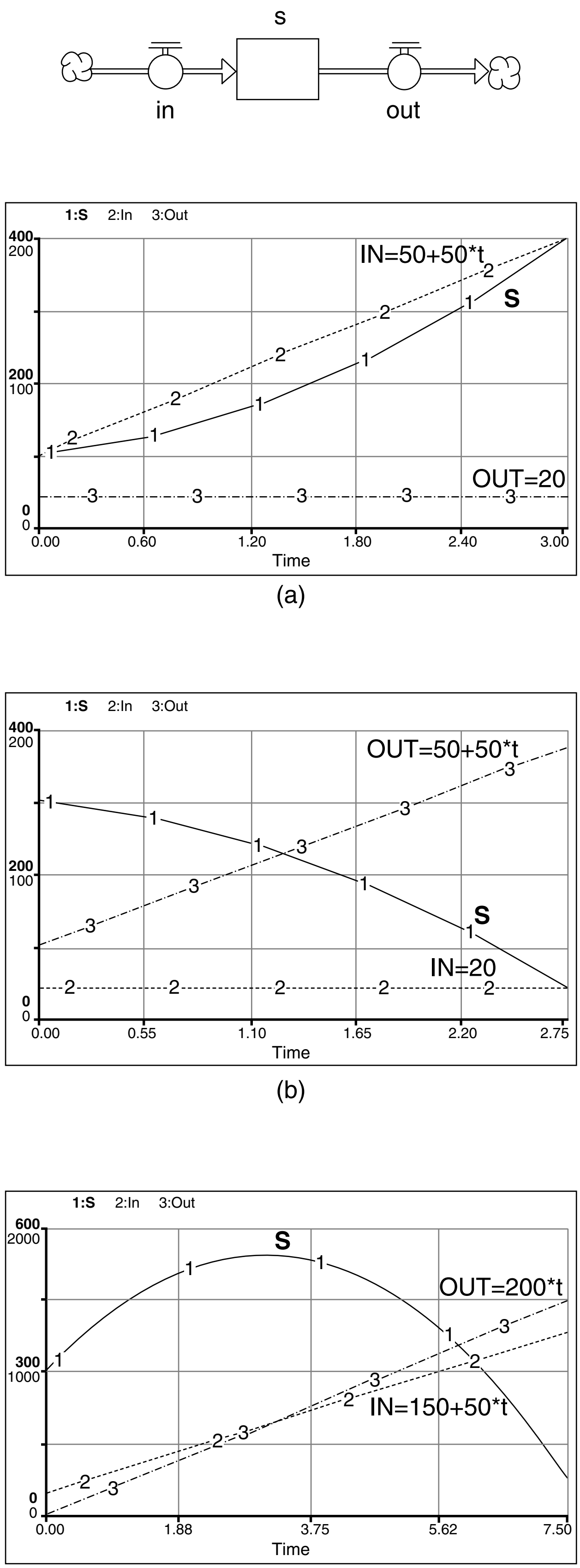


Figure 6. Dynamics of a stock (a) when the inflow is increasing linearly (outflow constant), (b) when the outflow is increasing linearly (inflow constant) and (c) when both flows change linearly and cross

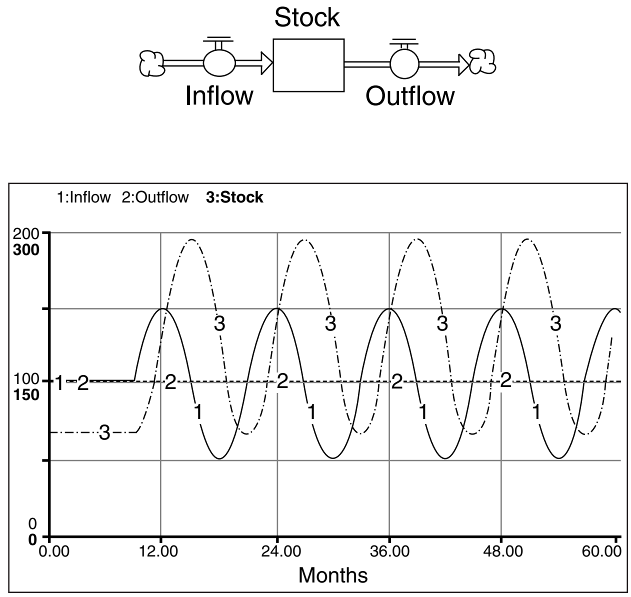
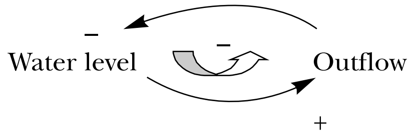


Figure 7. Inflow is oscillating and the outflow is constant. Observe that the oscillations in the stock are lagging behind the inflow oscillations, with a phase lag of $\pi/2$: integration creates delays

be approximately proportional to the water level, so the diagram would be:



In the above example, there is no conscious goal seeking. The tank would empty itself out and it would "decay" gradually to zero level. (One could say that the "effective" or implicit "goal" of this system is to reach zero water levels).

Positive and negative feedback loops are basic building blocks of dynamic structures. In reality, many such loops interact together. The feedback loops in interaction are displayed together in *causal loop diagrams.* (See for instance Figure 4 (b)).

4. DYNAMICS OF BASIC FEEDBACK STRUCTURES

4.1. Single linear positive feedback loop

As mentioned above, positive feedback loops create exponential growth or crash. Figure 8 (a) illustrates the simple linear positive feedback structure and its behavior. This model is of course solvable (already solved in the beginning of this article). The solution equation is:

Pop (*t*) = *Pop* (0)**Expon* (*BirthFraction*t*), *BirthFraction* a constant.

A very important property of the exponential curve is that it doubles its value in constant time intervals. The formula for the *doubling time* can be derived to be: $Td = \ln 2/(BirthFraction) \approx 0.70/BirthFraction$. Thus, once the growth constant (*BirthFraction*) is given, the process doubles itself every *Td*, independent of *Pop* (0). The doubling time in Figure 8 (a) is approximately fourteen time units. Figure 8 (b) is a less common version of linear positive feedback loop. It illustrates how positive feedback can lead to crash rather than growth. Consider the value of a stock in the stock market and assume that there is a critical value C, below which panic selling starts. Furthermore, in panic selling, the more the value drops the more people sell, the more it drops, and so on. We use the following flow equation to describe this process:

Loss = *fract** (*C-Stock*), where *fract* is a constant per unit time and *C* is the critical value.

Observe that the resulting dynamic behavior is a crash or collapse, which is a decline at increasing rates. (Constant "doubling time" still applies, except that what doubles is the discrepancy (*C-Stock*), rather than the stock itself).

4.2. Single linear negative feedback loop

As mentioned earlier, negative feedback loops create goal seeking or balancing dynamics. Figure 9 (a) illustrates the atomic linear negative feedback structure and its decaying behavior. This linear model is easily solvable and the solution equation is:

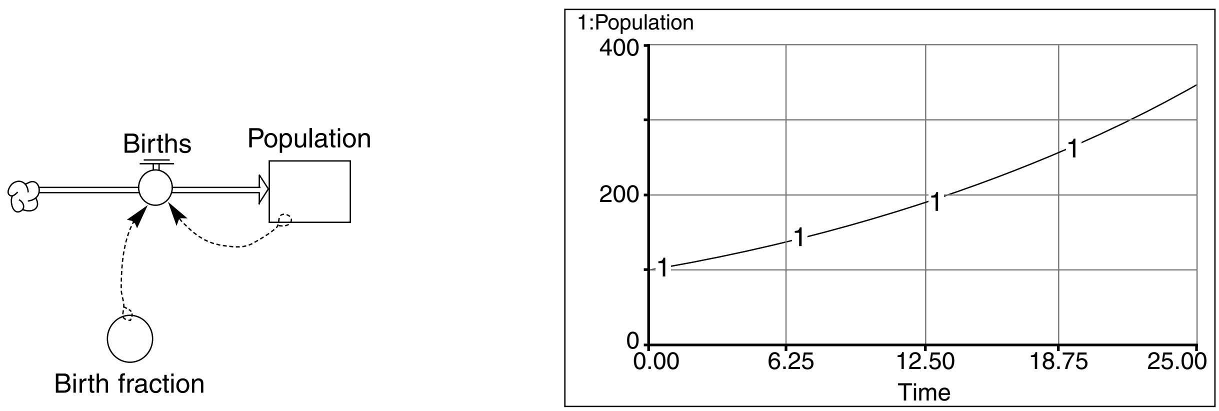
Pop (*t*) = *Pop* (0)**Expon* (– *DeathFraction***t*), *DeathFraction* a constant.

Analogous to exponential growth, an important property of exponential decay is that it *halves* its value in constant time intervals. The formula for the *half-life* can be derived to be: $Th = \ln 2/(DeathFraction) \approx 0.70/DeathFraction$. Thus, once the decay constant (*DeathFraction*) is given, the process halves itself every *Th*, independent of *Pop* (0). The half-life in Figure 9 (a) is approximately 7 time units. Figure 9 (b) illustrates another typical application of linear negative feedback loop. It is about the automatic heating/cooling of a room with a thermostat. There is a desired temperature and if the room temperature is below it, the thermostat activates the heater and if the room temperature is above the desired one then the thermostat activates the cooler. We use the following flow equation to describe this process:

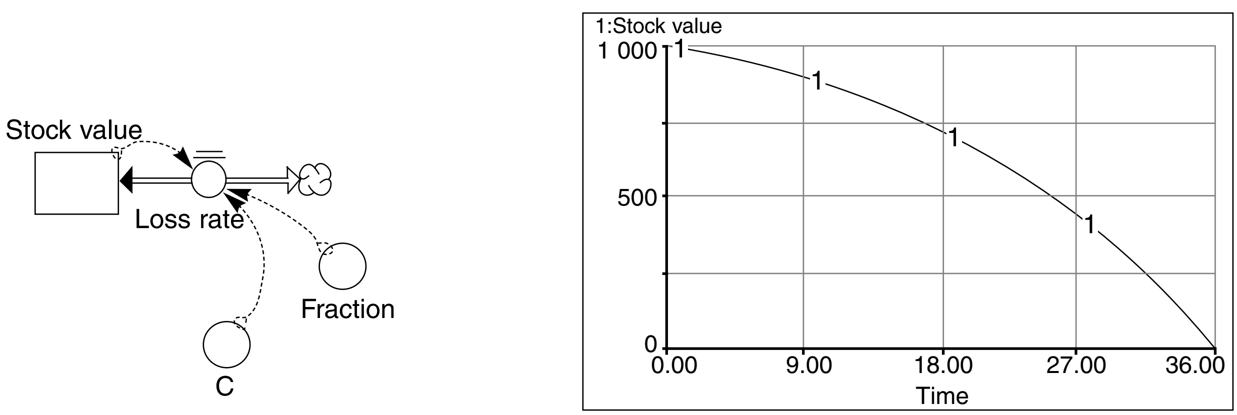
Temperature Change = *Discrepancy/Adjustment Time*,
where *Discrepancy* = (*Desired Temperature* – *Room Temperature*)

and *Adjustment Time* is a constant (in unit time) that represents how fast the thermostat reacts.

Observe that the resulting dynamic behavior is a goal seeking one. Whether the room temperature starts above or below the desired one, it gradually reaches it.



(a)



(b)

Figure 8(a) A simple linear positive feedback structure and resulting exponential growth; (b) A linear positive feedback example that would yield exponential crash

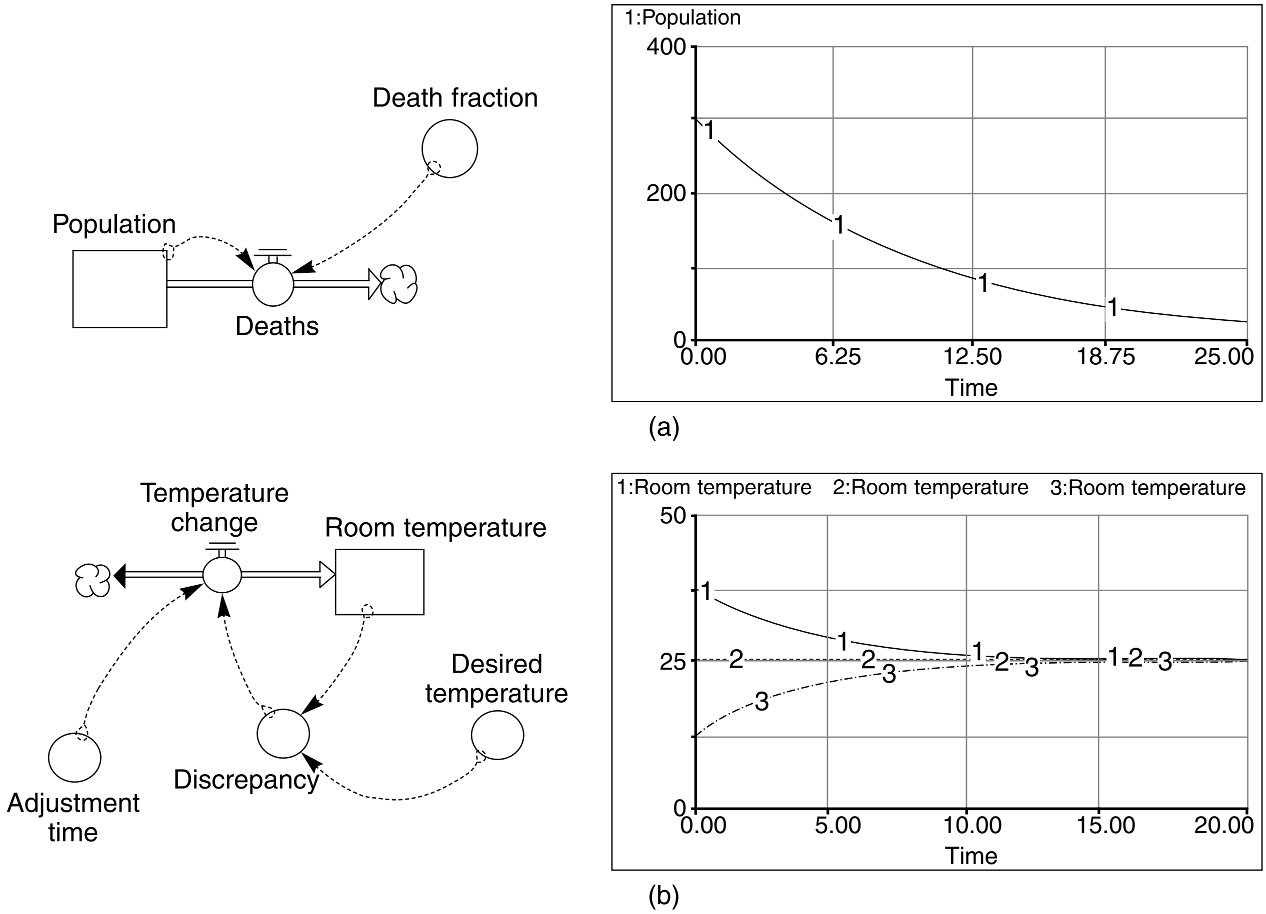


Figure 9. A simple linear negative feedback structure and resulting exponential decay; (b) A linear negative feedback example that seeks an explicit non-zero goal

Also note that the pattern of behavior is an exponential decay, characteristic of a negative feedback loop. (The "decay" is less obvious when the room temperature starts below the desired one, but in both cases what decays exponentially in time is the discrepancy = desired – room temperature). Finally, constant "half life" still applies, except that what halves is the discrepancy (desired – room temperature), rather than the temperature itself. (The adjustment time in these runs is 4 time units, so that the half-life is 0.70/0.25 ≈ 2.8 time units).

4.3. Simple coupling of linear loops

Linear positive loop with a constant outflow

When a model is subject to an external (independent) force (such as external temperature, rain, interest rates, competitor price), the model is said to be "non-autonomous" (or non-homogeneous). In these two sub-sections, we examine what happens to dynamics of linear feedback structures when they are non-homogeneous. When a positive loop is subject to a constant outflow, there are three possible dynamics. (Figure 10(a)). First, if the outflow *K* is equal to *initial inflow* = *fract*S* (0), in this trivial case the stock remains forever at the initial equilibrium *S* (0). Second, if the constant outflow *K* < *fract*S* (0), then it is also straightforward to guess that the dynamics will be an exponential growth, as the inflow will become greater and greater compared to the outflow. Finally, if the outflow *K* > *fract*S* (0), then we obtain somewhat less obvious dynamics: The positive feedback loop cannot create growth; instead, it yields a crashing behavior: the decline in a stock subject to a constant outflow is linear; but when positive feedback loop is involved, the decline becomes an exponential crash, not an immediately intuitive result.

Linear negative loop with a constant inflow

A similar case is a negative loop subject to a constant inflow. (Figure 10(b)). Again, if the inflow *K* is initially equal to the outflow = *fract*S* (0), then the variables stay at equilibrium. Second, if the inflow *K* is greater than the initial outflow *fract*S* (0), then the Stock increases by definition. But as it grows, its outflow becomes larger and larger, eventually equating the inflow *K*, reaching an equilibrium *Se* (*fract*Se* = *K*, yielding *Se* = *K/fract*). Finally, if the inflow K is less than the initial outflow *fract*S* (0), then the stock decays gradually to a non-zero equilibrium, again at *Se* = *K/fract*.

Coupling of linear positive and negative loops

When linear positive and negative loops are coupled, the resulting behavior will be either exponential growth, or decay, depending on which loop is "stronger" (*dominates*). An example of such a structure was already

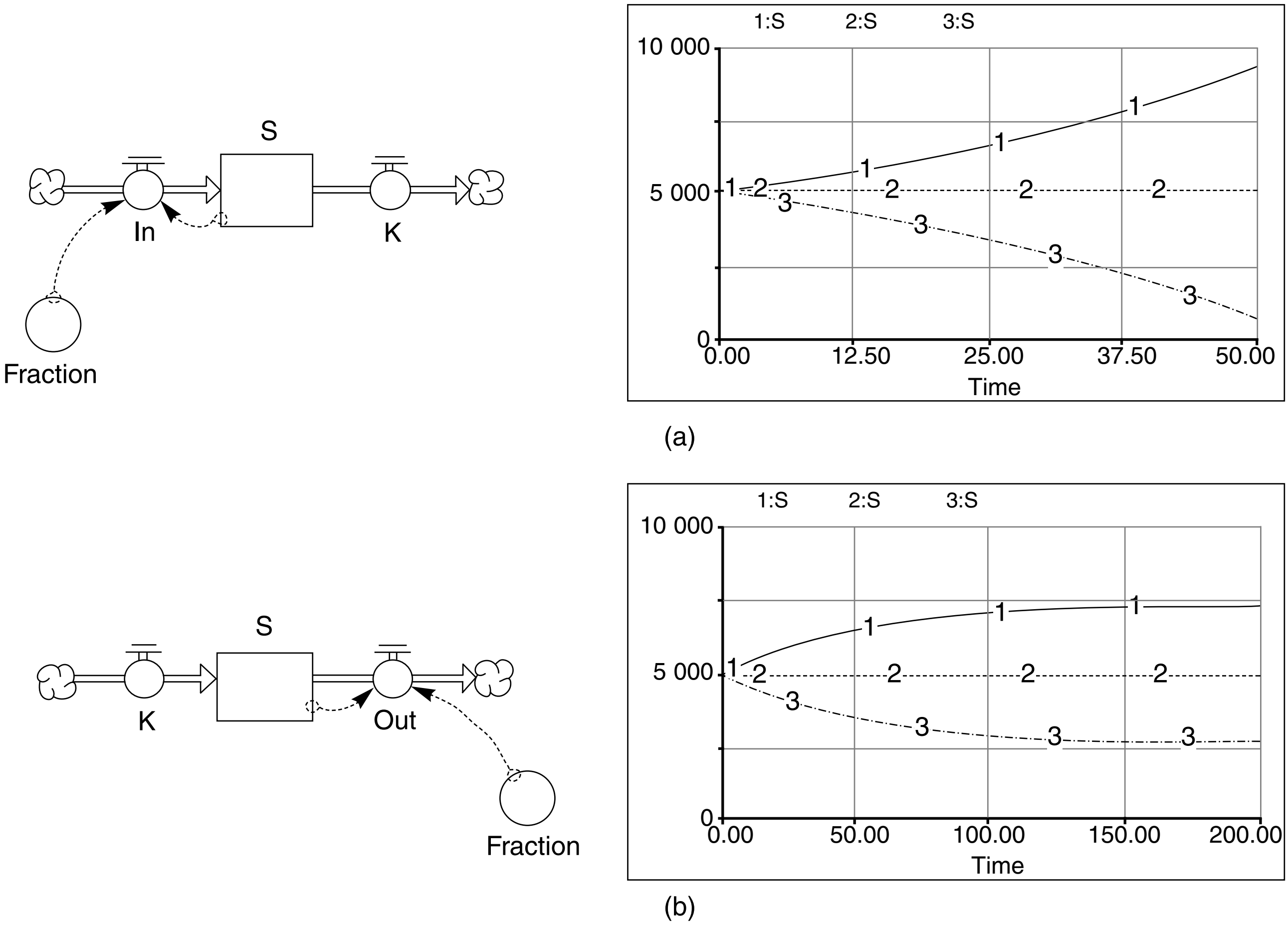


Figure 10. (a) Positive feedback structure subject to a contant outflow and possible behaviors; (b) Negative feedback structure subject to a constant inflow and possible behaviors

shown in Figure 3 (b) and the resulting possible behaviors in 3 (c). If the positive loop dominates (birth fraction larger than death fraction), the system yields exponential growth, if the reverse is true, then exponential decay and if the two fractions are equal, population remains of course unchanged. (Because of "superposition" property of linear models, combining linear structures cannot create novel behaviors).

4.4. A basic non-linear coupling of loops: density-dependent growth

The paragraph above shows that the linear coupling of feedback loops is not very subtle. A more interesting, *non-linear* coupling of positive and negative loops is quite common in different disciplines such as ecology, marketing, and epidemiology. Figure 11 illustrates this structure in the context of population dynamics. Assume that births are greater than deaths so that the population is growing. In linear coupling, this exponential growth would continue indefinitely. This is of course not realistic, as all growth must eventually face some limits (area, food, water, resource, air. . .). To model such a limiting process, we define "crowding" as population/maximum capacity. The maximum capacity (often called "carrying capacity" in ecology) is an aggregate summary measure representing various limits (area, food, etc), so that a given region cannot sustain more animals (or people) than capacity. Thus, "crowding" has a negative effect on birth fraction as seen in Figure 11, introducing a new feedback loop: Higher population means increased crowding, which lowers the birth fraction and which in turn creates a negative influence back on population: Population→ Crowding → Birth fraction → Population loop is a balancing one; it suppresses indefinite growth of population. As crowding gets larger and larger, birth fraction keeps falling and eventually approaches death fraction, stopping the population growth. The equations of the model are:

Deaths = *Death fraction*Population*, where *Death fraction* is constant (*per time*)
Crowding = *Population/Capacity*
Birth fraction = *f* (*Crowding*)
Births = *Birth fraction*Population*

The model is *non-linear* because *Crowding* = *Population/Capacity*, *Birth fraction* = *f* (*Crowding*), which means *Births* = *f* (*Population*)**Population*, a non-linear flow formulation.

In the simple version of density-dependent model, Capacity is assumed constant. The standard convention in these models is to specify the *birth fraction* = *f* (*Crowding*) function such that when *Crowding* = 1.0, the function yields *birth fraction* = *death fraction*. Since the death fraction is assumed constant, this specification is not difficult. A general birth fraction function is shown in Figure 11. This function says that when *Crowding* is rather low, increasing it would not have much effect on *birth fraction*. So, as *Crowding* becomes high enough to disrupt the basic needs of the population, *birth fraction* becomes more and more affected by it. Note finally that the function is specified so that it yields *birth fraction = death fraction* (0.06), when *Crowding* = 1.0. (The simpler formulation of Birth fraction is linear: *Birth fraction = a – b*Crowding*, a linearly decreasing function of *Crowding*. But note that this does *not* make the model linear!)

The dynamic behaviors of this model are shown in Figure 11. In these runs, Capacity is specified as 200, so that population eventually seeks this level, regardless of where it starts. If the *Population* starts above 200, *Crowding* is greater than 1.0, yielding a *birth fraction* smaller than *death fraction*, thus *population* declining gradually toward 200. If *population* is below 200, then *birth fraction* > *death fraction*, so there is growth. Note that there are two different growth possibilities. The most informative is the case when *Population* (0) is low enough, yielding an early exponential growth. It means that the net growth rate (births-deaths) is getting bigger and bigger. But as the population reaches a certain level (of *Crowding*), note that the net growth rate (births – deaths) starts becoming smaller and smaller (although it is still positive). Thus, as *Crowding* gets larger, the difference between births and deaths starts diminishing. This is the effect of the non-linear negative feedback loop described above. Hence, due to this non-linearity, the population growth in a density-dependent model can have two different phases: an exponential growth phase, followed by a goal-seeking growth phase. The first phase is caused by the *dominance* of the simple positive loop (Population → Births → Population) and the second phase is caused by the *shift of loop dominance* to the negative density dependence loop. Owing to this non-linear structure, the density-dependent model can generate the famous *S-shaped* (or *logistic)* growth dynamics. The significance of this dynamics is that it is possible for a process to start with exponential growth, but then later shift to goal-seeking behavior, as certain limits are approached. This type of dynamics can be seen in the spread of new innovations (see "R&D, technological innovations and diffusion," EOLSS on-line, 2002*)*, new ideas; certain epidemic dynamics, and various ecological processes. (Note finally in Figure 11, that if the initial population is already near enough its capacity, then the exponential growth phase does not exist at all. The growth is pure goal seeking from the start, because birth fraction is already dropping fast and approaching death fraction).

4.5. Beyond S-shaped growth: overshoot-then-decline

In the basic density-dependent structure above, the limiting capacity was assumed to be constant (such as fixed land). In many situations, the limiting factor is itself some resource stock (food) that the growth variable (population) depletes. Figure 12 illustrates such a situation where the population is consuming a food resource and the death rate of the population itself depends on the food availability (*Food per Capita*). In this example, as an illustration, the density-dependent effect is on death fraction (instead of birth fraction as in the previous example). Thus, *Population → Food per Capita → Death fraction → Population* loop acts as a density-dependent limit, just as in the previous example. What is new in this case is the fact that the "capacity" (*Food*) is a variable, being depleted by *Population*. (The specific functions *Effect_of_Food_on_Df* and *Effect_of_Food_on_Consumption* will be discussed in the following section. At this point, just note that *Death Fraction* is a decreasing function of *Food per capita* and *Consumption* is an increasing function of *Food per Capita* as well as *Population*). As the population grows exponentially, the consumption rate increases so much that at some point it causes a collapse in the food stock (Figure 13). Shortly after the food stock collapses, the population collapses too, due to increased death rates caused by starvation (as measured by *Food per Capita*).

The complete set of equations and functions are listed in Figure 13. Some of these equations will be discussed later in the following section. A couple of observations about the dynamics of this model: First, growth-then-collapse behavior would not have been possible in a single-stock model. This is mathematically provable, but also intuitively deducible: if there were a single stock, once the stock reached its peak and a slope (rate of change) of zero, then by definition, it would stay there, at its equilibrium. There needs to be some other source of energy (like another stock food not yet at its equilibrium) to push this stock away (down) from its own equilibrium. It is thus true in general that first-order, continuous time; autonomous systems can exhibit only *monotonous* dynamics. Second, this population-food model may generate other behaviors (decline only or oscillations) with other parameter and functions. For instance, it may yield declining population from start, if the initial *Population* is high enough relative to initial *Food*. Or, it may yield oscillations for different parameter values and/or functions. (see "Ecological interactions: predator and prey dynamics on the Kaibab plateau," EOLSS on-line, 2002).

Overshoot-then-collapse is observed in various sectors and problems in real life. (For a business application see "Market growth, collapse and failures to learn from interactive simulation games," EOLSS on-line, 2002). Another example that can yield overshoot-then-decline behavior is the dynamics of epidemics. Assume a contagious disease that can spread by contact between two groups of people: Susceptible and Infected. The causal loop and stock-flow diagrams are shown in Figure 14. A simple formulation for infection rate is obtained by basing it on "all possible contacts" (*Susceptible*Infected*):

*Infection Rate = InfectFract*Contacts,* where *InfectFract* represents what fraction of contacts results in infection

*Contacts = ContactFract*Susceptible*Infected,* where *ContactFract* represents what fraction of all possible contacts actually occurs per time unit (month).

The other flows are simple: There is an outflow from the Infected population (*Removal*), representing both deaths and recoveries. (It is assumed that recovered people gain permanent immunity, hence they do not

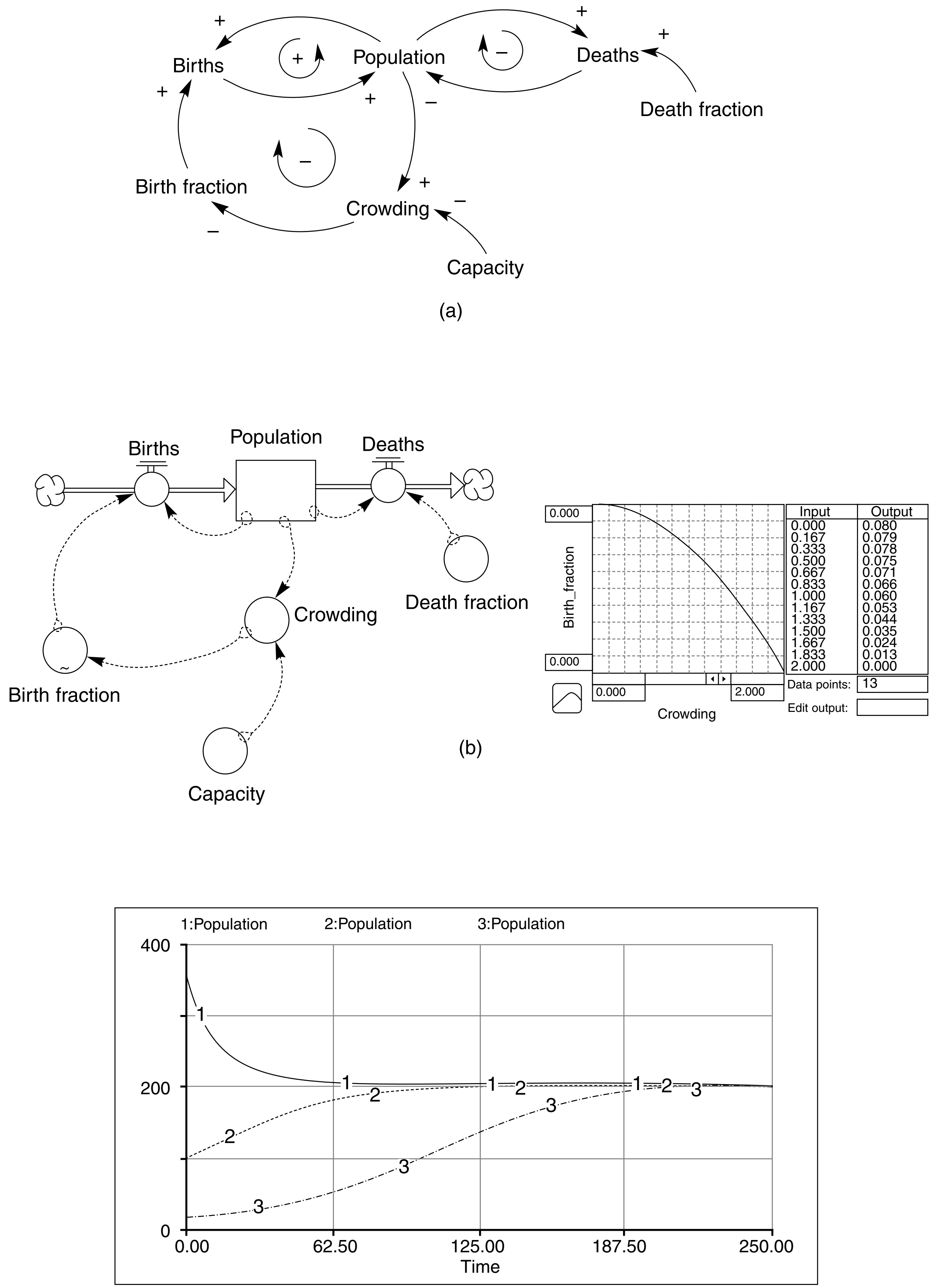


Figure 11. Density-dependent growth: (a) Causal loop diagram, (b) stock-flow structure and (c) possible dynamic behaviors

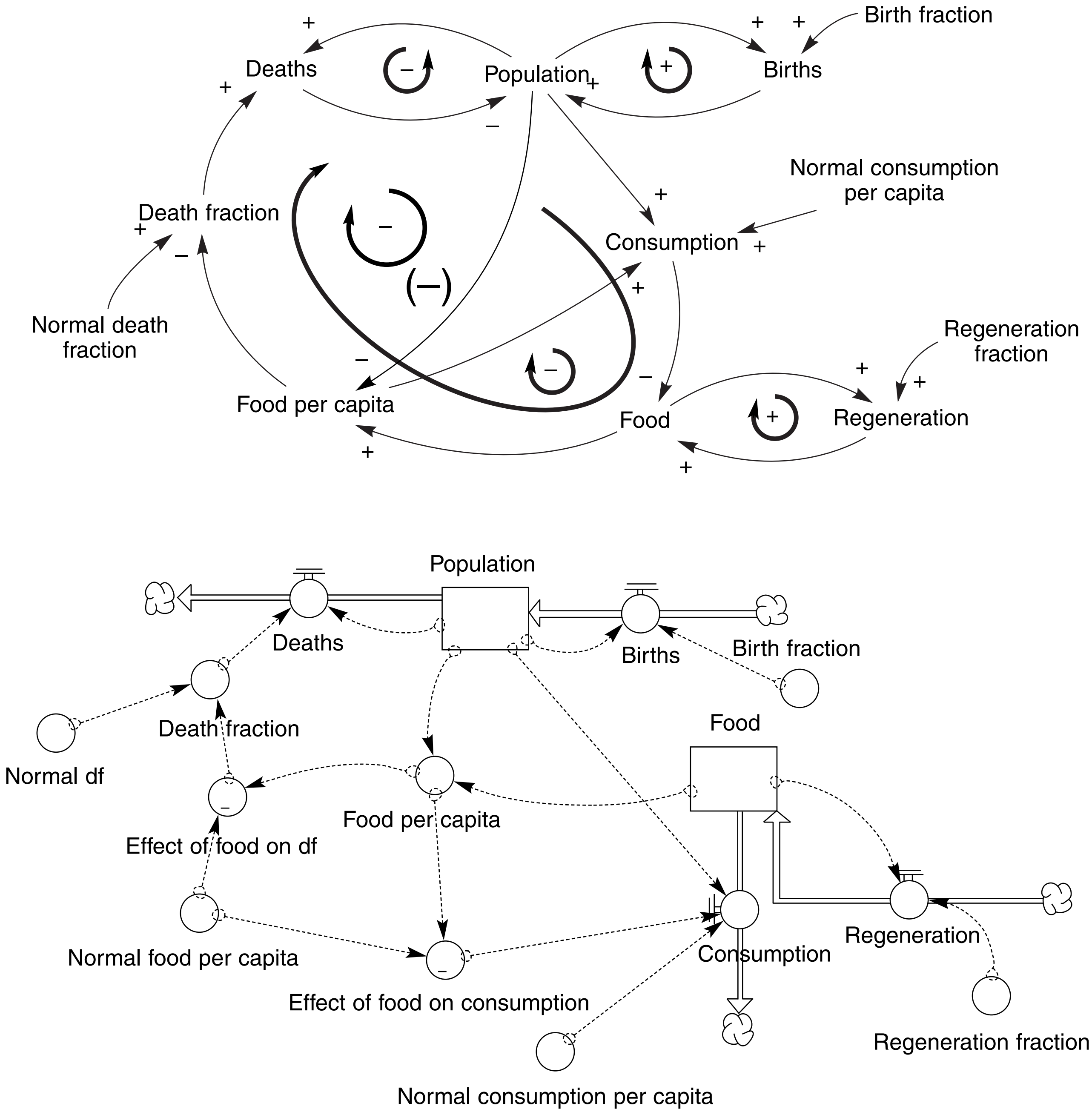


Figure 12. A population-food interaction model: causal loop and stock-flow diagrams

flow back into the susceptible pool). Finally, it is assumed that the susceptible group has a constant inflow (such as a net immigration). When the model is run with *ContactFract* = 0.1 per month and *InfectFract* = 0.01, Infected group exhibits the overshoot-then-collapse behavior seen in Figure 14 (c). Thus, in this case there is an epidemic outbreak, which eventually settles down to an equilibrium level (of a relatively few number of people). The equilibrium is obtained when all flows equal (*In* = *Infection Rate* = *Removals*). In a second experiment, when the model is run with a lower infectivity (*InfectFract* = 0.002), then we observe "epidemic oscillations." With even lower infectivity values and/or higher initial values of the infected stock, there would be no epidemics at all – *Infected* would decline monotonically over time.

5. FORMULATION PRINCIPLES AND GENERIC MODEL STRUCTURES

5.1. General principles of formulation

System dynamics models, being causal-descriptive ones, consist of causal cause-effect equations. As such, these equations must obey some fundamental principles:

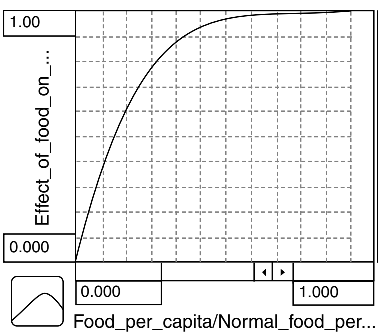
Equations must have real-life meaning

Equations must correspond to real processes. They must have defendable meaning. It is unacceptable to use an equation just because it fits data or provides excellent predictions. (This point was already discussed above, in the section on causality versus correlation).

Equations

Food (t) = Food (t - dt) + (Regeneration – Consumption) * dt
INIT Food = 2 000
Regeneration = Regeneration_fraction * Food
Consumption = Effect_of_food_on_consumption * Normal_consumption_per_capita * Population
Population (t) = Population (t – dt) + (Births – Deaths) * dt
INIT Population = 50
Births = Birth_fraction * Population
Deaths = Death_fraction * Population
Birth_fraction = 0.2
Death_fraction = Effect_of_food_on_Df * Normal_Df
Food_per_capita = Food/Population
Normal_consumption_per_capita = 15
Normal_Df = 0.05
Normal_Food_per_capita = 45
Regeneration_fraction = 0.5

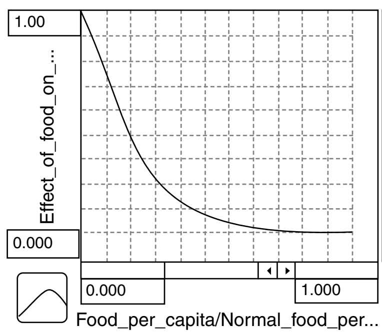
Effect_of_food_on_consumption



Input	Output
0.000	0.000
0.091	0.365
0.182	0.620
0.273	0.775
0.364	0.875
0.455	0.930
0.545	0.980
0.636	0.990
0.727	0.995
0.818	0.998
0.909	1.000
1.000	1.000

Data points: 12
Edit output:

Effect_of_food_on_Df



Input	Output
0.000	9.990
0.091	7.450
0.182	4.950
0.273	3.350
0.364	2.400
0.455	1.800
0.545	1.300
0.636	1.150
0.727	1.070
0.818	1.020
0.909	1.010
1.000	1.000

Data points: 12
Edit output:

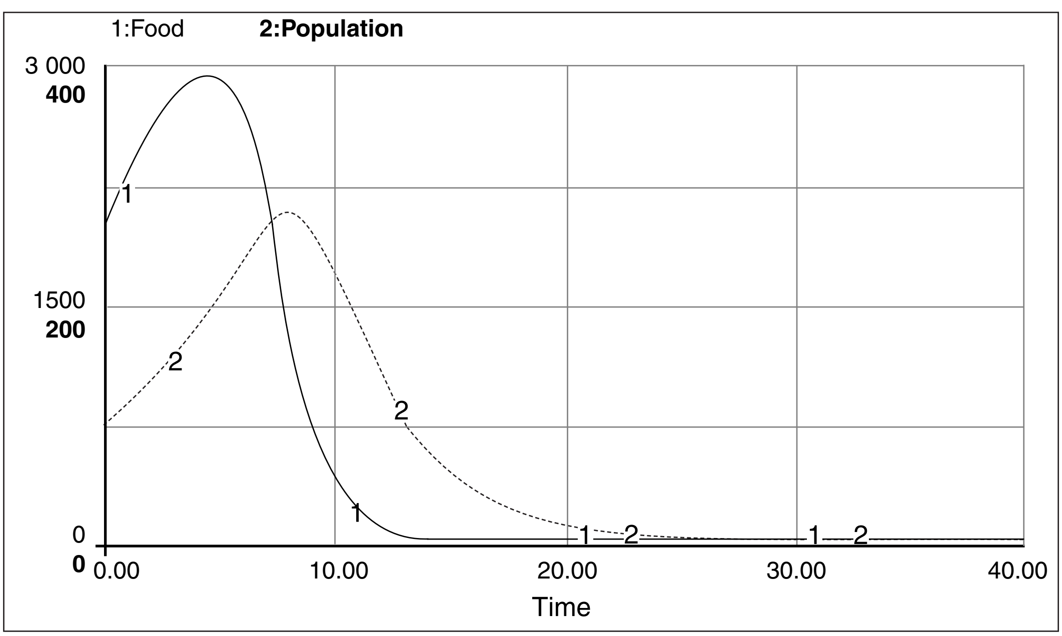


Figure 13. The population-food interaction model: Equations and the resulting overshoot-then-collapse behavior

For an equation to be meaningful, all of its variables and parameters must of course be meaningful. Variables and parameters must correspond to real-life concepts. This includes quantitative variables as well as qualitative ones and it includes applied variables as well as theoretical ones.

Equations must be dimensionally consistent

If the units of the right hand side variables in a birth rate formulation yield something like 1/people, this is of course unacceptable (units in reality are people/time period). It is equally unacceptable to use a variable like "population effective" formulated as $100/\text{population}^{1/2}$ and then multiply it by some "coefficient K" with units $\text{people}^2/\text{month}$, so as to yield birth rate in people/month! The *dimensional consistency* of each equation must be verified, *without* including any "dummy" coefficients in order to make the units consistent. The real-life units of each variable and coefficient must be plugged in and the units of the right hand side must "automatically" yield the units of the output variable, without any "dummy" scaling coefficient.

Equations must yield valid results even in extreme conditions

When the population is assumed zero (a hypothetical extreme), death rate must also be zero, for we know this is necessarily the case in real life. If the death rate formulation yields a non-zero value under the zero population tests, arguing, "zero population is an unrealistic extreme" is not a valid defense. When an equation fails under extreme conditions (very small, very large, zero, and so on), this often indicates that there is something illogical in the formulation that is probably yielding wrong results in the normal operating range of the input variables, but in more "hidden" ways.

Realism should not be sacrificed for mathematical simplicity

The model must provide a realistic description of the real processes with respect to the dynamic problem. Typically, real processes involve *non-linear* relations, interacting *multiple feedback loops* and time *delays*. Each of these three factors greatly complicates the mathematical analysis of the model. A tendency in such cases is to approximate non-linearities with linear ones, to ignore the existence of some of the loops and to omit time delays (that increase the order and dynamic complexity of the model). None of these simplifying assumptions is justifiable just to obtain mathematical tractability. Mathematical (analytical) exactness is not of primary concern in system dynamics; much more essential is the realism of the model. Since solutions in system dynamics method are obtained by simulation, non-linearities need not be regarded as a threat to tractability and therefore should not be evaded.

Equations must not unrealistically assume optimality or equilibrium

Equations must describe the way the real world works, not the way it "should" work assuming that all actors behave optimally, or assuming that variables are at

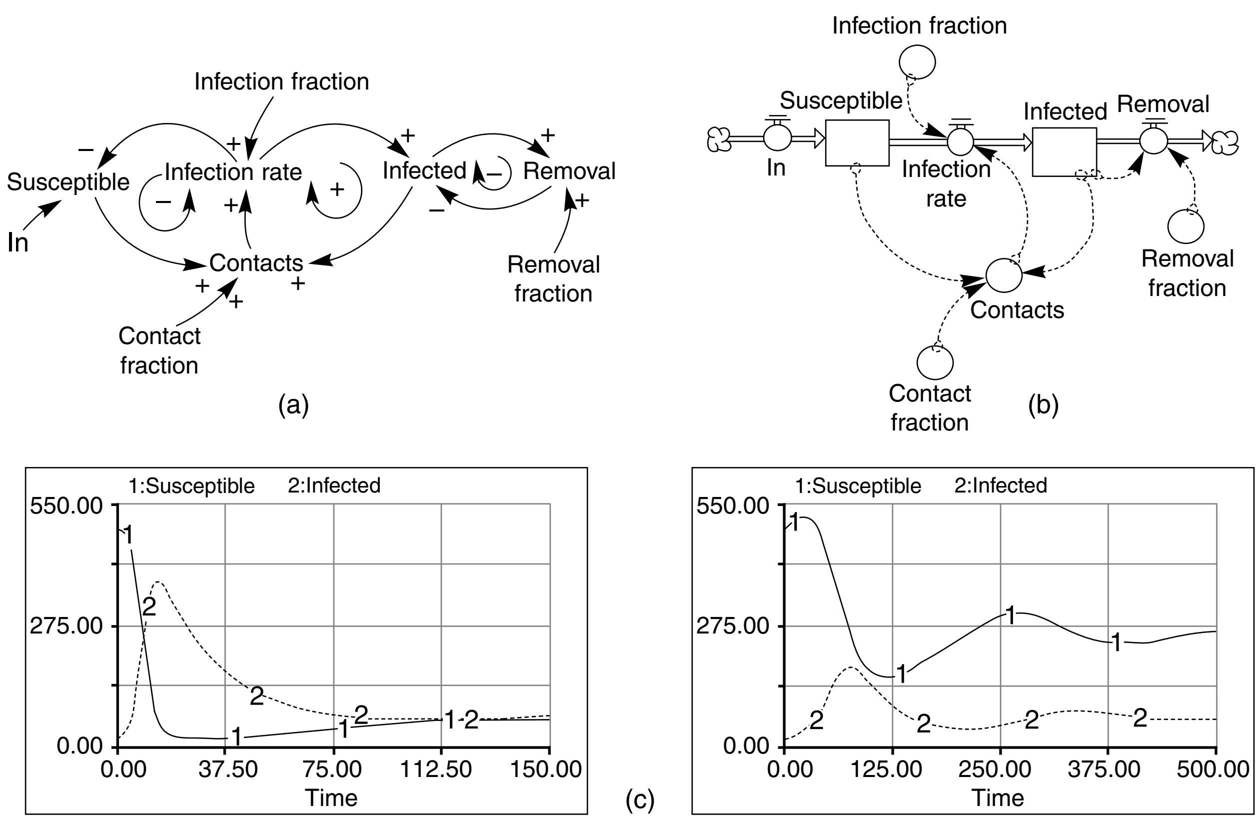


Figure 14. Epidemic dynamics: (a) causal loop diagram, (b) stock-flow structure and (c) possible dynamic behaviors

equilibrium. It is not realistic to assume that most decision-makers have access to all data and algorithms needed to solve the complex optimization problems. Nor is it realistic to derive equations based on the assumption that variables are at equilibrium. The purpose of system dynamics modeling is to investigate the causes of dynamic behavior and equations derived from equilibrium (i.e. non-dynamic) assumptions do not normally apply in the dynamic mode of operation.

5.2. Linear formulation

A linear equation assumes that the output is proportional to the input. The general form is $Y=a+bX$, where the intercept a and slope b are both constant. Typical examples we have already seen include:

*Births = birthfract*Pop*
*Recovery = Recovfract*Infected*
Crowding = Pop/Capacity
TempChange = (DesiredTemp – Temperature)/AdjustTime

The above formulations are all linear, because *birthfract, Recovfract, Capacity,* and *AdjustTime* are all assumed constant. If any of these parameters were functions (direct or indirect) of the stock variable (*Pop, Infected, Temperature*), then the formulation would be *non-linear,* because proportionality (or constant slope) assumption between input and output would not hold. Thus, the simple definition of a non-linear formulation is "a formulation that is not linear." This means any mathematical expression other than $a+bX$ (including any x^a, $\ln(x)$, e^x and any combination of such functions). Furthermore, in a formulation involving more than one variable, any product or division of variables would be non-linear as well, since this would violate proportionality. In sum, linearity is a pretty restrictive assumption. In some situations such as the ones mentioned above it may be reasonable to assume linearity, but it should be kept in mind that these situations are rare in systemic feedback models, as will be seen below.

5.3. Non-linear equation formulation

Prevalence of non-linearity

In a dynamic feedback model, non-linearity is often "natural," almost a rule. Consider the simplest density-dependent population model seen earlier. We have:

Crowding = Pop/Capacity and
*BirthFract = a – b*Crowding*

Note that both of these equations are linear, since we assume that *Capacity, a* and *b* are all constant. But when we next write *Births = Birthfract*Pop,* we obtain a *non-linear* model. Non-linearity comes from the fact that *Births = f (Pop)*Pop.* What is interesting here is the fact that we simply write down a succession of linear equations, but obtain a non-linear model when the feedback loop is "closed." The recognition of *Population → Crowding → BirthFract → Births → Population* loop automatically makes the model non-linear. That is why; most feedback models are non-linear by their very nature. To make a dynamic model linear, one must not only assume proportional relations between all pairs of variables, but must also further omit most of the feedback loops operating in the system.

Non-linear formulation examples and techniques

A basic and standard non-linear formulation we have already seen is the "product" formulation used in the epidemic dynamics model:

*Infection Rate = InfectFract*Contacts*
*Contacts = ContactFract*Susceptible*Infected, ContactFract* and *Infectfract* are constants

The Infection flow in the above model is non-linear, as it is a function of the product of the Infected and Susceptible stocks. This formulation has wide applicability, for instance in formulating how a new product spreads between people by word of mouth. (see "R&D, technological innovations and diffusion," EOLSS on-line, 2002). More generally, this type of formulation is used when the existence of a flow requires the simultaneous existence of two different stocks. For instance, in a predator-prey relation, the rate preying would be *f (Predators*Preys)*. This says that the preying rate must be zero if there are no preys or if there are no predators. (For an actual formulation see "Ecological interactions: predator and prey dynamics on the Kaibab plateau," EOLSS on-line, 2002).

Multiplicative and additive "effect" formulations

A generalization of the above product formulation is the "multiplicative effect" formulation. The general form of the formulation is:

Y = (Effect of $X1$ on Y)*(Effect of $X2$ on Y)*. . .*(Effect of Xn on Y)*Ynormal

The above equation states that Y is a multiplicative function of $X1$, $X2$,. . .Xn and the effects of these variables are formulated by "Effect of Xi on Y" functions times the "normal" value of Y, Ynormal. Thus, for each effect, we have:

Effect of Xi on $Y = f(Xi/Xi,\text{normal})$, so that the input variables are normalized.

The meaning of "normal" values is that when all input variables are at their normal values, we expect the output value Y to be at its normal value Ynormal. (These are also called "reference" values). In a multiplicative effect formulation this requires that all $f(.)$ must yield 1, when Xi are at their normal value Xi,normal. In short, $f(1) = 1$ for all multiplicative effect functions. Such a *normalization* is important in building robust models, because if absolute values are used as inputs to these functions, then it would be almost impossible to experiment with different model parameters, as the input values would quickly go outside the ranges of the functions.

Examples of such effect formulations involving single input variables were already seen in the population-food interaction model above (see Figure 13). We have:

DeathFract = (Effect of Food on *Df*)**NormalDf*
NormalDf=0.05
Effect of Food on *Df* = *f* (*FoodperCapita/NormalFoodperCapita*)
NormalFoodperCapita=45 (kg/animal)

The above formulation says that when the input variable *FoodperCapita* is at its "normal" value of 45, the death rate is at its "normal" value of 0.05. Thus, *f* (1)=1 by definition. (see the function plotted in figure 13). The specific shape and values of the function depends of course on the specifics of the problem. The function plotted in Figure 13 states that when there is more food than "normal" the death rate does not get any lower than normal. (The *x*-axis is therefore cut off at 1). Furthermore, the function assumes that when *FoodperCapita* is moderately below its normal value (between 0.75 and 1), death fraction is not affected much by food availability. (The function starts yielding values increasingly greater than one, only after the food per capita ratios drops below 0.75).

This formulation is a single-input multiplicative effect formulation. The formulation is non-linear in two ways. First, since *Deaths* = *DeathFract*Pop* and *DeathFract* = *f* (*Food*), then in effect we have: *Deaths* = *f* (*Food*)**Pop*, a non-linearity due to the product of two state variables. A second source of non-linearity is the fact that *DeathFract* is itself a non-linear function of *Food* (the curve in Figure 13).

In the same model, a second similar multiplicative formulation exists:

Consumption = (Effect of Food on *Consumption*) **NormalConsumptionperCapita*Population*
NormalConsumptionperCapita = 15 kg/animal/month
Effect of Food on *Consumption* = *f* (*FoodperCapita/NormalFoodperCapita*)

This formulation is very similar to the previous one. The function must again satisfy *f* (1)=1. (Figure 13). It says that when there is "normal" or above normal food, the consumption per capita is at its "normal" (15 kg/animal/month). If the food per capita ratio drops significantly (below about 0.60), then per animal food consumption starts decreasing at increasing rates. One minor technical difference between this example and the previous one is that since there is only a single input, the effect formulation is embedded in the flow equation, as a short cut.

More typical applications of effect formulation would involve several input variables. For instance, consider sales personnel booking orders. Each sales person has a "normal" sales productivity, but this may be affected by at least two obvious factors: Price of the product and the advertising done by the company. Thus, we have:

Sales = *Sales Productivity*Sales Force*
Sales Productivity =(Effect of Price on *SP*)*(Effect of Advertising on *SP*)*Normal *SP*
Effect of Price on *SP* = *f* (*CompanyPrice/ CompetitorPrice*)
Effect of Advertising on *SP* = *f*(*CompanyAdvertising/ CompetitorAdvertising*)
Normal *SP* = 50 units/person/month
SalesForce, Company Price, CompanyAdvertising, typically endogenous variables
CompetitorPrice, CompetitorAdvertising, could be variables or constants

The functions *f* (*CompanyPrice/ CompetitorPrice*) and *f* (*CompanyAdvertising/ CompetitorAdvertising*) are shown in Figure 15. Observe that they obey the principles outlined above. They both satisfy *f* (1)=1. Thus, the Sales Productivity (*SP)* formulation above says that when the price is at its normal value *and* the advertising is at its normal value, sales productivity is at its normal value of fifty. Note that in this problem context, "normal" values are defined as *competitor price* and *competitor advertising*. In other words, we assume that there is a major competitor and if our price is at about their level, then the price has "no effect" on sales productivity. Similarly, if our advertising (Liras/month) is about at competitor's level, then it has neutral effect on sales productivity. If our price is above the competitor price, then it has the effect of reducing our sales (function value less than 1). And if our advertising is below the competitor's, it has similar negative effect. (See both functions in Figure 15a). Conversely, if our price is below the competitor's, then the price effect function is greater than 1, meaning positive effect on sales productivity. And if our advertising is above the competitor's, then the advertising effect function is greater than 1, meaning positive effect on sales productivity. (See Figure 15a). In the context of the larger model involved, sales people, company price, and advertising are most likely variables, all taking feedbacks from sales, revenues, salaries, cost-profit ratio, and so on. Furthermore, competitor price and/or advertising may be variables too in a model with larger boundary. Thus, it is important to note that all the "normal" values used in effect formulations may be variables as well as constants. Finally, there can be more effects such as "effect of delivery delay" and so on, yielding *Sales Productivity*=(Effect of Price on *SP*)*(Effect of Advertising on *SP*)*(Effect of *DelivDel*)*Normal *SP*, where the Effect of *DelivDel* function would again be similarly constructed.

Multiplicative effect formulations have two important properties. First, in a multiplicative formulation, extreme values of any of the inputs (if yielding zero) will completely *dominate* the outcome. Thus, the previous example assumes that if our price were three times higher than the competitor price then nobody would buy our product (*Sales Productivity* would be zero), even if we did a lot of advertising. The same may be true if our delivery delay were five times as high as the competitor's, and so on. In such cases, if only one of the effects is at its extreme worst value, then the output would be at its worst value, regardless of how favorable the other inputs are. This assumption is realistic in some cases, but not so in others, as will be seen

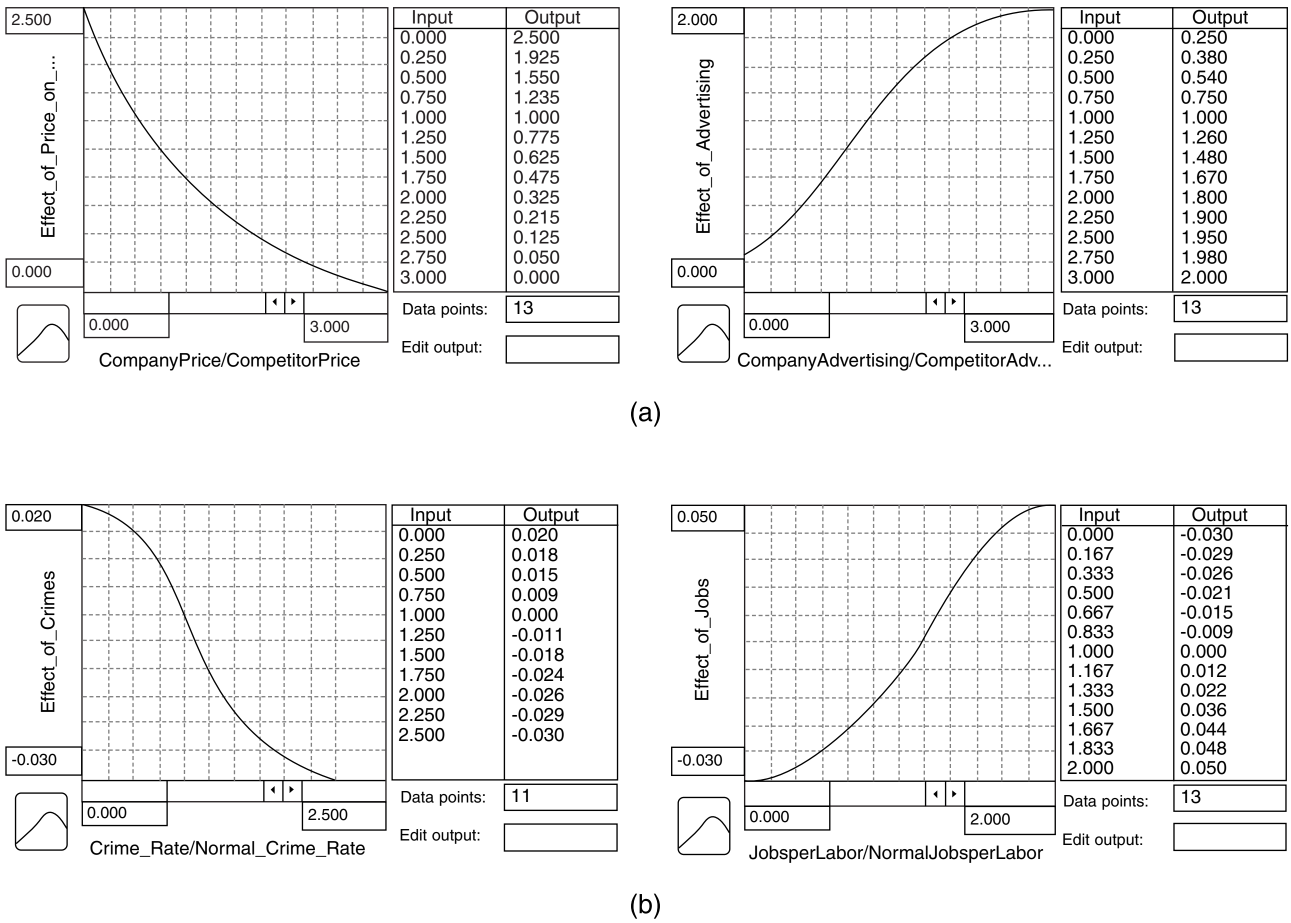


Figure 15. (a) Multiplicative and (b) Additive effect function examples

later. A second important characteristic of multiplicative formulation is, since effects are multiplied, the combined effect will increase or decrease geometrically, quickly yielding very small or large values, if several inputs are involved. If there are eight multiplicative effects and each has a moderately unfavorable value of 0.7, then the combined effect would be 0.7^8 (0.05), an extreme reduction in the output. It is therefore risky to use more than four to five multiplicative effects, because controlling and validating such unstable multivariable functions is extremely difficult.

It is therefore desirable in some cases to use *additive* effect formulations. Consider the urban population dynamics problem discussed briefly earlier. The causal loop diagram is shown in Figure 4. According to this conceptual model, in/out migration fraction depends on job availability, house availability and crime rate. The hypothesis is that high job and house availability values and low crime rate would encourage in-migration, whereas low job and house availability values and high crime rates would induce out-migration. We will give an additive effect formulation to represent this migration process. Since housing and jobs have very similar effects, we will illustrate the formulation by focusing on jobs and crime rate. The following equations present such a formulation:

*In_outMigration = MigrFract*Population*

MigrFract = *NormalFract* + (Effect of *Jobs*) + (Effect of *Crimes*)

NormalFract = constant {can be 0 or some positive or negative fraction}

Effect of *Jobs* = *f* (*JobsperLabor*/ *NormalJobsperLabor*)

Effect of *Crimes* = *f*(*CrimeRate*/ *NormalCrimeRate*)

NormalJobsperLabor, *NormalCrimeRate*, typically constants

JobsperLabor, *CrimeRate*, other endogenous variables.

The *In-outMigration* formulation is a "bi-flow" which means that if *MigrFract* is positive we have in-migration and if *MigrFract* is negative we have out-migration. In the above additive formulation, *NormalFract* would be set 0, if we assume that when the jobs and crime rate are at their "normal" values, there is normally no in-migration to or out-migration from the city. (Alternatively, if *NormalFract* was say – 0.05 per year, then we assume that even if jobs and crime rate are at their normal values, there are some other not-included factors such as schools, which would create out-migration from the city).

The functions *f* (*JobsperLabor/NormalJobsperLabor*) and *f* (*CrimeRate/NormalCrimeRate*) are shown in Figure 15(b). Note that their properties are different than the multiplicative ones. First, note that in an additive formulation "effects" have units, 1/time in the above example, while they are dimensionless in multiplicative formulation. Second, at their "normal" values they yield 0 rather than one: *f* (1) = 0. This property is necessary for the addition to return *NormalFract*, when both *JobsperLabor* and *CrimeRate* are at their normal values. Third, the additive effects can (often must) have negative values, since they add to or subtract from a "normal" value. Thus, the *MigrFract* formulation above says that when *JobsperLabor* is above some given "normal" *JobsperLabor*, then it has the effect of promoting in-migration (function returning a positive fraction). And if *CrimeRate* is below *NormalCrimeRate*, it has similar positive effect on in-migration. (See both functions in Figure 15b). Conversely, if *JobsperLabor* is below *NormalJobsperLabor*, then the jobs effect function is less than 0, meaning a push for out-migration. And if *CrimeRate* is above *NormalCrimeRate*, it has similar negative effect. (See Figure 15b). Note that there can be *no domination* of a single factor in such an additive formulation. Even if *CrimeRate* had an extremely negative effect, if *JobsperLabor* was very high, the effects could cancel each other, or people could even migrate-in, according to the additive formulation. In a multiplicative formulation, it is possible to shut off completely the in--migration if *JobsperLabor* is extremely low, regardless of how low the crime rate is. This would be impossible in an additive formulation. On the other hand, observe that the above "bi-flow" (in or out) formulation of migration would be impossible with multiplicative effects. If the multiplicative effects are allowed to become negative, then one would get meaningless results if both effects are negative: product of two negative effects would yield a positive effect, hence creating an in-migration! (Also, when one effect is negative and the other one is positive, in a multiplicative formula, the larger is the positive effect, the more negative would be the combined effect – again meaningless). In practice, whether multiplicative or additive effect formulation is suitable depends on the specifics of the problem. (Also note that in an additive formulation "effects" have units, 1/time in the above example, while they are dimensionless in multiplicative formulation).

5.4. Time delay formulations

Time delays often play an important role in the dynamics of systems. Significant time lags may intervene between causes and their effects. There are two general categories of time delays in system dynamics: *material* delays and *information* delays.

Simple material delay

Such delays exist on conserved (material) stock-flow chains. For instance, orders do not immediately create arrivals of goods; this delay is often called "lead time" of orders. When new workers are hired, there is a training period before they join the actual production force. Sometimes delays are in the very definition of variables: normally, four years must pass before freshers can graduate from university. Two material delay examples are shown in Figure 16(a). In the first example, it takes an average of three years (Seedling Delay) before seeds planted can create seedlings. In the second example, there is a production delay before a production decision can create finished goods. Observe that in both cases delays exist on a conserved (material) stock-flow chain. The generic stock-flow diagram is also shown in Figure 16(c). Since delays involve stock-flow structures, they introduce phase lags between the inputs and outputs. (Figure 7 provides a clear visual illustration of how a stock in between two flows introduces a phase lag). There are different ways of formulating such delays, depending on the specifics of the problem. For instance, in one extreme, when there is a change in the inflow, it is assumed that there is no change at all in the output for $t < Del$ and then the entire change is observed at $t = Del$. In other words, $OUT\ (t + Del) = IN(t)$. This is called a *discrete* delay and assumes that the delay *Del* is very rigid and applies to each entity in the process exactly. The assumption is proper in relatively micro-level modeling. In more macro-level models, the delay is an average, with a rather large variance. The process involves an aggregation of many different types of entities so that some may be processed much faster than *Del* and some much slower than *Del*. In these cases, under the assumption of a *continuous* and *homogeneous* mixing of the different entities, a suitable formulation of the outflow is:

$OUT = Stock/Del$

Thus, the above formulation says that the inflow *IN* accumulates in the stock and then we expect 1/*Del* of the stock to flow out per unit time. If it takes an average of three years for seeds to yield seedlings, we expect Seeds/3 seedlings per year. In the production example, the average Finishing rate is (*Goods_In_Process*)/*ProdDelay*. In the above formulation, when the inflow *IN* is suddenly stepped up or down by a constant quantity, the output *OUT* gradually changes toward the new value of *IN*. (We assume that initially, $IN=OUT$, so that the structure is initially at *equilibrium*). Note that *OUT* is changing continuously, before *Del* as well as after *Del*. (Figure 16(b)). The dynamic behavior of *Stock* (and *OUT*) is a goal seeking, negative exponential curve:

$$Stock\ (t) = Del*IN + (Stock(0) - Del*IN)*EXPON(-\ t/Del),$$

$$OUT = Stock/Del$$

The term $Del\ *IN$ is the *new equilibrium* value of the stock, so that when $Stock_e = Del\ *IN$, the outflow $OUT_e = Del*IN/Del = IN$, which is consistent at the equilibrium.

Observe in Figure 16 (b) that *Del* is just an average delay. Some entities are processed much faster and some are processed much slower. Mathematically, *Del* is the average value of the above exponential function. When $t = Del$ is plugged in, it can be seen that 63 percent of the discrepancy between the initial value and the new equilibrium is covered and 37 percent remains. (In a discrete delay, the entire discrepancy would be covered suddenly at $t = Del$ and no action at all before or after *Del L*). The continuous, exponential delay formulation is suitable in most system dynamics problems, since these macro-level models typically involve aggregation of many different entities involving different time constants. But one major drawback of this simple exponential delay is the fact that when *IN* is changed, the largest response in *OUT* is obtained in the next instant in time and the shape of the *OUT* curve has a monotonically decreasing derivative (in magnitude). In some cases, it is more realistic to have no or very small response early in the process and increasingly larger responses later on (an S-shaped response). This type of delay behavior – somewhere in between the simple exponential delay and the simple discrete delay – can be obtained by *higher order* exponential delay formulation, as will be seen below.

Simple information delay

In some situations, a variable influences another one with a delay, but not in a material stock-flow chain. This may mean "delayed effect" of a variable on another one, such as *crowding* affecting birth fraction after a time delay. Or it may represent human awareness, gradual learning of a changing situation, such as "perceived jobs per labor" as a delayed information about real jobs per labor (in which case our migration decision would be affected not directly by the latter, but by the "perceived" one). Two information delay examples are shown in (a) of Plate 6.63–1. The first example illustrates the "delayed effect" implementation. In the density-dependent population growth model, *Crowding* had a negative effect on *Birth Fraction* (Figure 11). Now suppose that a change in *Crowding* level does not immediately affect *Birth Fraction*, but it does so after some time delay. Thus, *Crowding* directly affects "Implied Birth Fraction" and then after some time delay, this latter becomes "Actual Birth Fraction." The corresponding stock-flow diagram is shown in (b) of Plate 6.63–1. *ImpliedBirthFract* is an input to the information delay structure and the delayed output is the *ActualBirthFract*. Just as in the material delay case, this can be represented either as a discrete delay, exhibiting no effect at all for a period of *Del* and then the entire effect at *Del* or as a continuous, gradual delay. In the information delay context, discrete effects are rare; continuous delay structure illustrated in (a) of Plate 6.63–1 is much more common. The equations corresponding to the first example in (a) of Plate 6.63–1 are:

ImpliedBirthFract = *f* (*Crowding*), exactly as given already in Figure 11

Correction = (*ImpliedBirthFract* – *ActualBirthFract*)/ *Correct_Del*

ActualBirthFract (*t*+*dt*) = *ActualBirthFract* (*t*) + *dt** (*Correction*)

In the above formulation, the *INPUT* to the delay structure is *ImpliedBirthFract* and the delayed *OUTPUT* is *ActualBirthFract*. The dynamic behavior of this continuous information delay structure is exactly the same as that of the continuous material delay shown in Figure 16 (b). Initially, we assume that the system is at equlibrium i.e. *ActualBirthFract* = *ImpliedBirthFract*, so that the correction flow *Correction* = 0. As a test, when the *INPUT* (*ImpliedBirthFract*) is stepped up or down, the *OUTPUT* (*ActualBirthFract*) approaches the *INPUT* in an exponential goal-seeking manner. (See Figure 16 (b)). At *t*= *Del*, 63 percent of the discrepancy between the initial value and the new equilibrium (the new *INPUT*) is again covered.

The second example in (a) of Plate 6.63–1 is an illustration of gradual human awareness, learning or perception. Assume that a certain product has a varying quality (measured by frequency of breakdown) that customers would like to know in deciding to purchase the product or not. Customers in the market place cannot instantaneously learn the actual level of the product quality; they can only learn it gradually, after a perception (learning) delay. Thus, the *INPUT* of the information delay is *Product Quality* and the *OUTPUT* is *Perceived Quality*, which would lag behind the actual *Product Quality*. The exact behavior of *Perceived Quality* depends on the value of the *perception delay (Percept_Del)*. The smaller the *perception delay* the faster will be the *Correction*; hence faster will be the tracking of the *INPUT*. But the risk of using too small "smoothing" constants is that they may yield unstable behavior if the *INPUT* changes too fast. Note that the information delay structure given in Plate 6.63–1 is an application of the *goal-seeking* structure shown in Figures 9 and 20. In the information delay structure, the "goal" is the *INPUT* of the delay structure and the *OUTPUT* stock seeks this goal. (see the generic structure shown in (b) of Plate 6.63–1). Finally observe that the information delay structure used above is a continuous version of the *exponential smoothing* method often used in data smoothing for forecasting. The *OUTPUT* equation for the generic structure given in (b) of Plate 6.63–1 is:

OUT(*t*+*dt*) = *OUT*(*t*) + *dt** *Correction*, where
Correction = (*IN* – *OUT*)/*SmoothDel*

Thus:

OUT(*t*+*dt*) = *OUT*(t) + *dt** (*IN*(*t*) – *OUT*(*t*))/*SmoothDel*

The standard exponential smoothing equation is:

OUT(*t*+1) = *OUT*(*t*) + *alfa**(*IN*(*t*) – *OUT*(*t*))

Comparing the last two equations, we see that they are the same by calling *alfa = dt/SmoothDel*. If our model is discrete (*dt*=1), then *alfa* is just 1/*SmoothDel*. The information delay structure is thus a continuous, exponential averaging process. The larger *SmoothDel* (the smaller *alfa)* the more smoothing is done on the *INPUT*. The noisy *INPUT* in (c) of Plate 6.63–1 illustrates the smoothing action. Note that *OUTPUT* is a smoother and time-lagged version of the *INPUT* and that the larger *SmoothDel*, the smoother the *OUTPUT* is.

Higher order delays

Delay structures are not direct causal formulations; they are *behavioral* approximations to the outputs of many cause-effect interactions that would make the model unnecessarily too complicated, if explicitly modeled. So, instead of modeling all the workforce, schedule and raw material interactions in a production setting, we simply approximate the resulting delayed behavior by a material delay formulation. As such, it is critical that the delay structure capture the

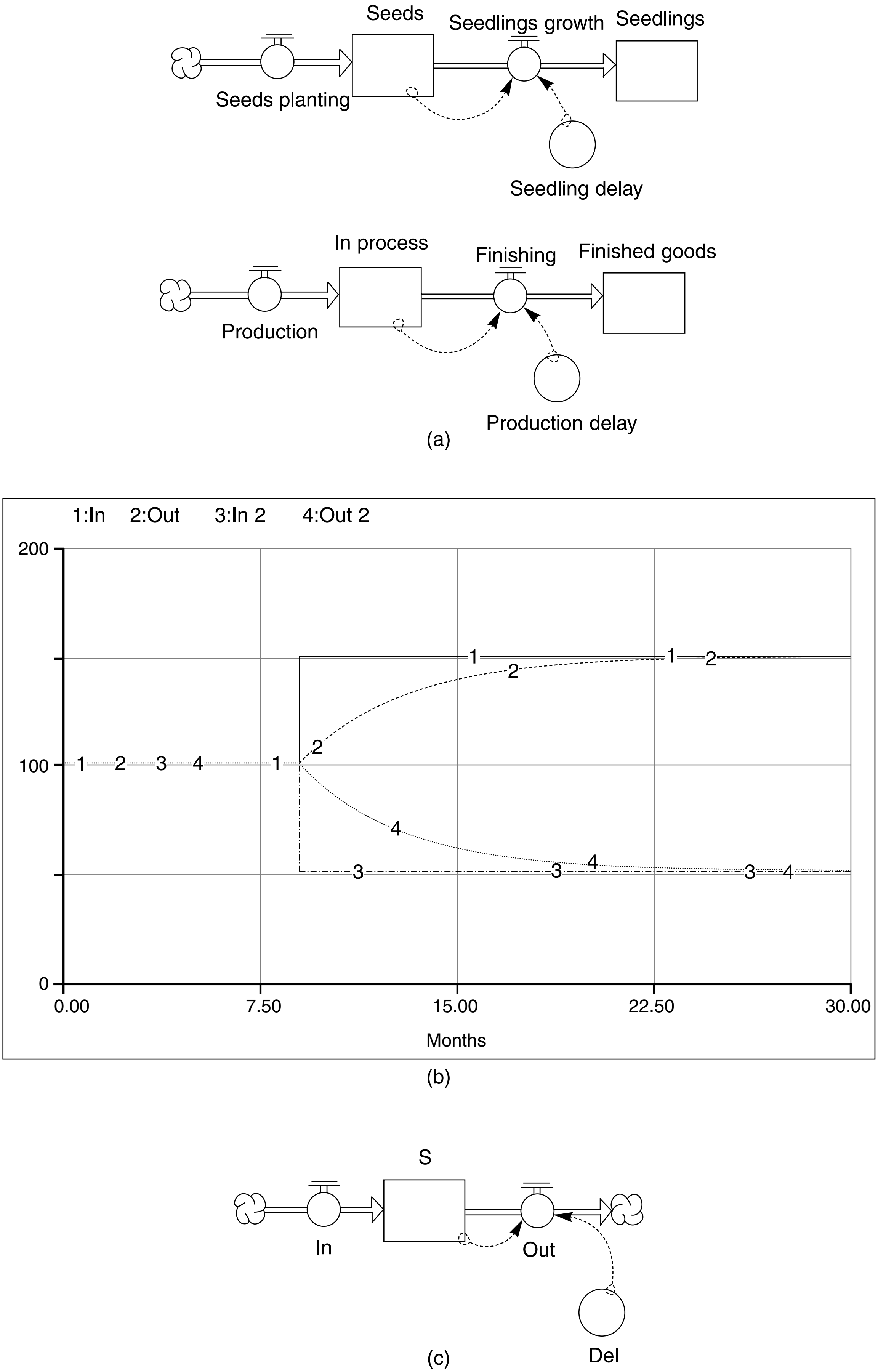


Figure 16. First order material delay (a) Examples, (b) Output behavior, (c) Generic structure

input-output dynamics of the real structure realistically. As mentioned earlier, a major drawback of the simple exponential delay is the fact that when *IN* is changed, the largest response in *OUT* is obtained in the next instant in time and has a monotonically decreasing derivative. In some cases, it is more realistic to have no or very small response early in the process and increasingly larger responses later on (an S-shaped response). This type of delay behavior can be obtained by *higher order* exponential delay formulation. A higher order delay is simply a serial cascading of several first order exponential delays. In Figure 17 (a), a third-order material delay example is illustrated, by simply extending the forest seed-planting example already introduced in Figure 16 (a). The generic structure of the third-order continuous material delay is given in Figure 17 (b). The equations are identical to those already described for the first order material delay. (Note that for the total average delay of the entire structure to be *Del*, the delay time used in each sub-structure must be *Del*/3). The dynamic behavior of the third-order delay structure, when disturbed from equilibrium is shown in Figure 17 (c). Observe that the output has now an s-shaped pattern – first increasing slope and then decreasing slope. The initial period of low response may be important in capturing the dynamic response characteristics of some real life processes. Similarly, a third order information delay is obtained by cascading three first order information delay structures, each with delay times of *Del* /3 (Figure 17(d)). The third order delay is a common illustration of high-order delays. Most software has built in macro functions for third order delays. If more steep S-shaped response is desired, one can easily increase the order of the delay by cascading additional first or even third order delay structures. In the limit, the discrete delay formulation *OUT* (*t*+ *Del*) = *IN*(*t*) can be shown to be the nth order continuous delay structure, as $n \to \infty$.

5.5. Delays in action: oscillating structures

An important consequence of having delays in structures is that they are potential sources of oscillatory behavior. For instance, consider the density dependent growth structure and its S-shaped behavior given in Figure 11. In the previous section, the following information delay structure was added to the density-loop:

ImpliedBirthFract = *f*(*Crowding*), exactly as given already in Figure 11

Correction = (*ImpliedBirthFract* – *ActualBirthFract*)/ *Correct_Del*

ActualBirthFract (*t*+*dt*) = *ActualBirthFract* (*t*) + *dt** (*Correction*)

With the addition of the above delay, the population now has the potential to oscillate up and down around the equilibrium (*Capacity*), before it eventually settles down. Because the effect of *Crowding* on birth fraction is "delayed," population can overshoot the *Capacity*, then undershoot, and so on.

The stock management structure

A very typical managerial application of delays is in the context of the standard goal-seeking (stock-adjustment) structure illustrated earlier in Figure 9. The generic version of this stock-adjustment structure is also provided in Figure 20. The equations are:

Discrepancy = *Goal* – *State*
ControlAction = *Discrepancy*/ *AdjustTime*
State (*t*+*dt*) = *State* (*t*) + *dt** *ControlAction*

The behavior of the above model is pure exponential goal seeking. But if delays exist anywhere around the goal-seeking loop, then the system has the potential to oscillate. The delay can be in measuring the State (an information delay between the *State* and the *Discrepancy*), between the *Discrepancy* and *ControlAction* (a delayed action, information delay) or between the *ControlAction* and the *State* (a material delay before the action reaches the *State*). One or more of such delays can cause the system to oscillate. For instance, an extension of this simple goal-seeking structure is the stock management structure, used in various management settings (such as inventory management, human resources management, cash management, driving, weight control. . .) We illustrate the general stock adjustment problem in the context of inventory order management in Figure 18. Compared to the basic goal-seeking structure, there are two major extensions: first, there is an outflow from the stock (*Shipments*) and this outflow must be taken into consideration in writing the ordering equation. Second, there is a material delay (*Supply Line*) before the ordered goods reach our stock, which must also be taken into account. (The *Supply Line* delay is formulated by the standard material delay equation: *Acquisition_rate* = *Supply_Line*/ *Acquisition _delay*).

First, if the outflow *Shipments* is not considered, it is easy to prove that the original equation *OrderRate* = (*DesiredStock* – *Stock*)**StockAdjFract* will not bring the stock to its desired equilibrium. At the equilibrium, we must have *OrderRate* = *Shipments*, which means:

(*DesiredStock* – *Stock*)**StockAdjFract* = *Shipments*, so that:
$Stock_e$ = *DesiredStock* – *Shipments*/ *StockAdjFract*, a value smaller than *DesiredStock*.

To correct this "steady state error" we simply add Shipments to the order equation:

OrderRate = (*DesiredStock* – *Stock*)**StockAdjFract* + *Shipments*,
which will yield $Stock_e$= *DesiredStock*.

But the above formulation is not realistic, as it assumes that we know the actual values of shipments at all times. A more realistic formulation must use "estimated" or "expected" shipments instead. This latter can be formulated by using the standard exponential smoothing (averaging) structure seen earlier:

Expected_shipments(t) = *Expected_shipments*(*t* – *dt*) + (*Correction*) * *dt*
Correction = (*shipments* – *Expected_shipments*)**alpha*
alpha = 0.5

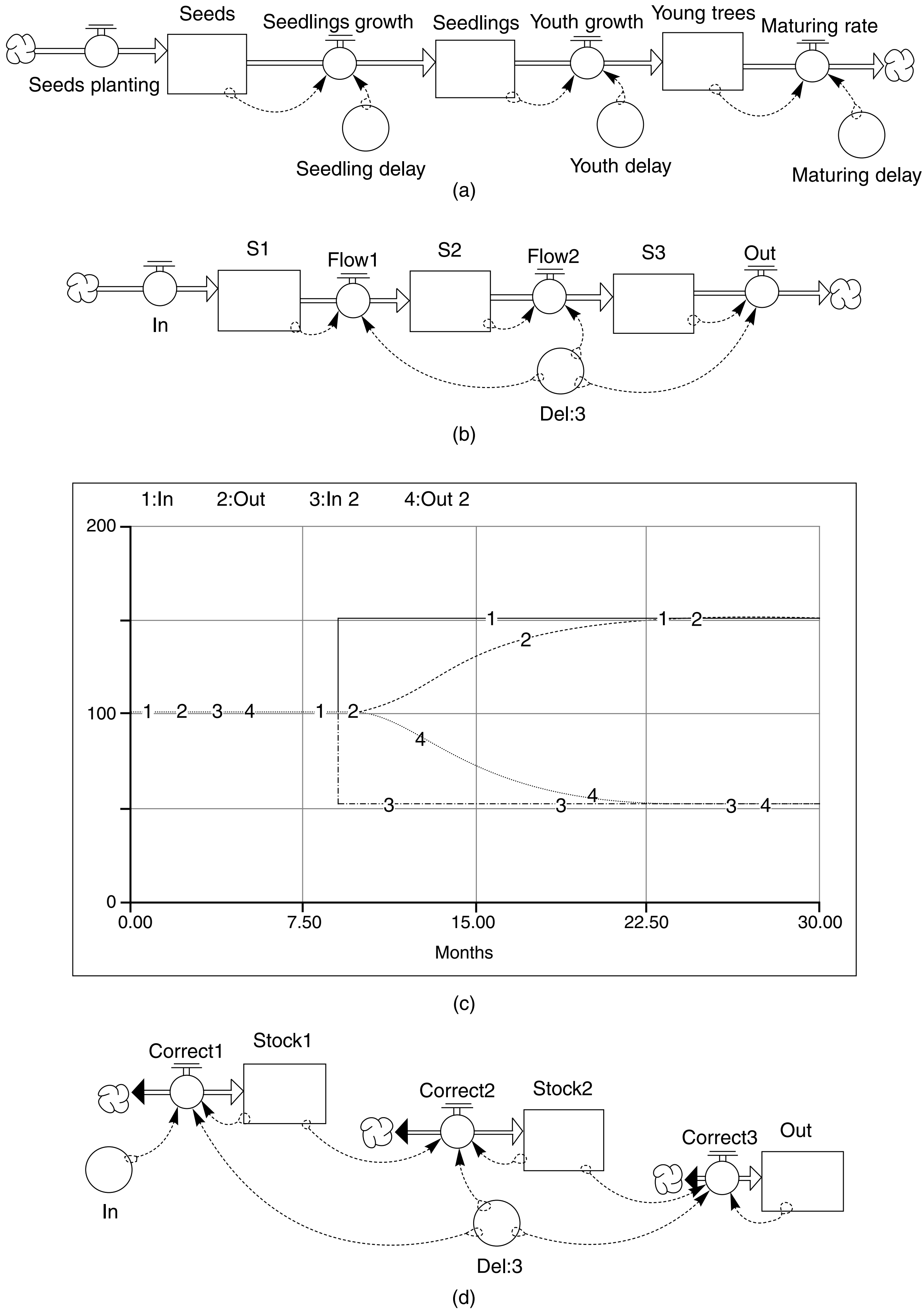


Figure 17. High (third) order delays: (A) Material delay example, (b) Generic third order material delay structure, (c) Output behavior, (d) Generic third order information delay structure

And the "desired" stock level can also be tied to the expected shipments, since higher shipments would normally call for higher desired inventory levels:

*Desired_*stock = *Stock_Level_Factor*Expected_shipments*
Stock_Level_Factor = 3

The resulting ordering equation is:

OrderRate = (*DesiredStock* – *Stock*)***StockAdjFract* + *Expected_Shipments*

When the model is run with this ordering formulation, the stock indeed reaches its desired level (of thirty), as seen in Figure 19(a). Observe though the oscillations in the inventory and supply line stocks. These oscillations are caused by the existence of the material delay (*Supply Line*) between orders and the inventory. The oscillations are further amplified by the fact that the above ordering formulation completely ignores the existence of the supply line delay. A better formulation would take into account the existence of *Supply line* (*SL*) as follows:

order_decision = *stock_adjustment* + *supply_line_adjustment* + *Expected_shipments*
desired_supply_line = *acquisition_delay*Expected_shipments*
supply_line_adjustment = (*desired_supply_line-supply_line*)***SL_adj_frac*
SL_adj_frac = 0.4

The only new term in the above formulation is "supply line adjustment" which is identical to the standard "stock adjustment" term defined earlier, except that "desired supply line" must be properly defined. At the equilibrium, the supply line must yield an outflow (acquisition) equal to Shipments. Thus,

Supply_line/Acquisition_delay = *Shipments*,

yielding

$Supply\_line_e = acquisition\_delay*Shipments$,

and since we do not exactly know the shipments, then:

$Desired\_supply\_line = Supply\_line_e = acquisition\_delay*Expected\_shipments$.

When the same model is run with the above ordering policy (all parameters remaining unchanged), the oscillatory behavior disappears as shown in Figure 19(b). Inclusion of "goods on order" (*supply line*) in the ordering equation prevents unnecessary over-ordering, hence yielding an improved, and more stable inventory system.

There are naturally other oscillatory structures. For instance, the epidemic model described in Figure 14 was demonstrated to exhibit oscillatory behavior for certain values of infection fraction. Also, when two stocks are coupled with a negative feedback loop (such as a predator-prey relation), there is potential for oscillatory behavior. (see "Ecological interactions: predator and prey dynamics on the Kaibab plateau," EOLSS on-line, 2002) The generic structure of the 2–stock oscillator is shown in Figure 20.

5.6. Generic structures and dynamics

Some important elementary structures are frequently used as basic building blocks in diverse applications. These are sometimes called "generic" structures. For convenience, the most basic generic stock-flow structures, their causal loop diagrams and their possible dynamic behaviors are summarized in Figure 20. Each of these structures was either described in detail or at least mentioned earlier in this article. Figure 20 is provided as a quick reference.

6. MATHEMATICAL AND TECHNICAL ISSUES

6.1. Solutions, equilibrium, and stability

There are no general mathematical solution methods for non-linear, high-order feedback models. In some situations, approximate or partial solutions can be obtained. But this often requires that the order of the model be reduced, some non-linear relations linearized, and some feedback loops broken, in which case there is always the risk of losing some of the most critical dynamic characteristics of the system. Analytical procedures are therefore very rarely applicable in system dynamics studies. A more useful and relevant application of such procedures is in examining the elementary generic structures mentioned above. Exact knowledge of the dynamic properties of these structures is very useful in building larger models and carrying out quasi-mathematical analysis.

A relatively more tractable mathematical problem is to determine the *equilibrium* solutions of the model and their *stability*. The equilibrium solutions are those that make all derivatives zero. In other words, for each stock i, the sum of its flows must be zero. The equilibrium solutions can be obtained by solving the following set of equations:

$\sum_j flows = 0$, for each stock *i*.

If there are *n* stocks, this would give *n* simultaneous non-linear equations to be solved for S_i. Although these equations are algebraic (not dynamic), solving non-linear algebraic equations is still a very difficult problem in general. In most cases, "some" of the equilibrium points can be found, but no knowledge of how many global equilibria there are. A very important property of an equilibrium point is its *stability*. There are different technical definitions of stability. For the purpose of this article it is enough to state that an equilibrium point is stable, if solutions that start near it tend toward it or stay in the close neighborhood of it. The stability of each equilibrium point can be analyzed by linearizing the model in the close neighborhood of each equilibrium point, and checking the eigenvalues. (see "Equilibrium and stability analysis," EOLSS on-line, 2002). Finally, some non-linear models can exhibit very complicated dynamic behaviors such as *limit cycles, bifurcations, period-doubling, strange attractors,*

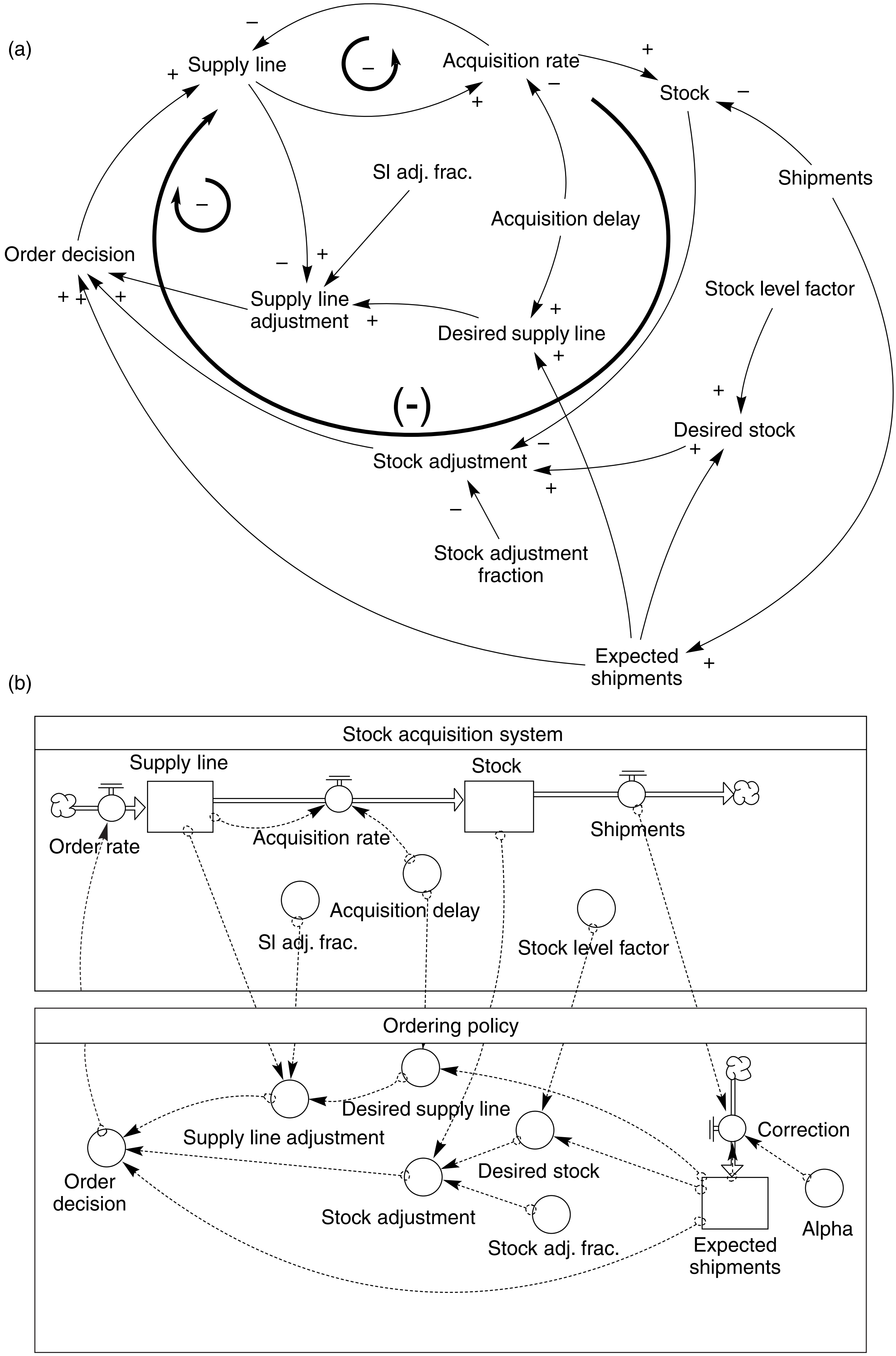


Figure 18 The general stock adjustment problem applied to inventory management. (a) causal loop diagram (b) stock-flow structure

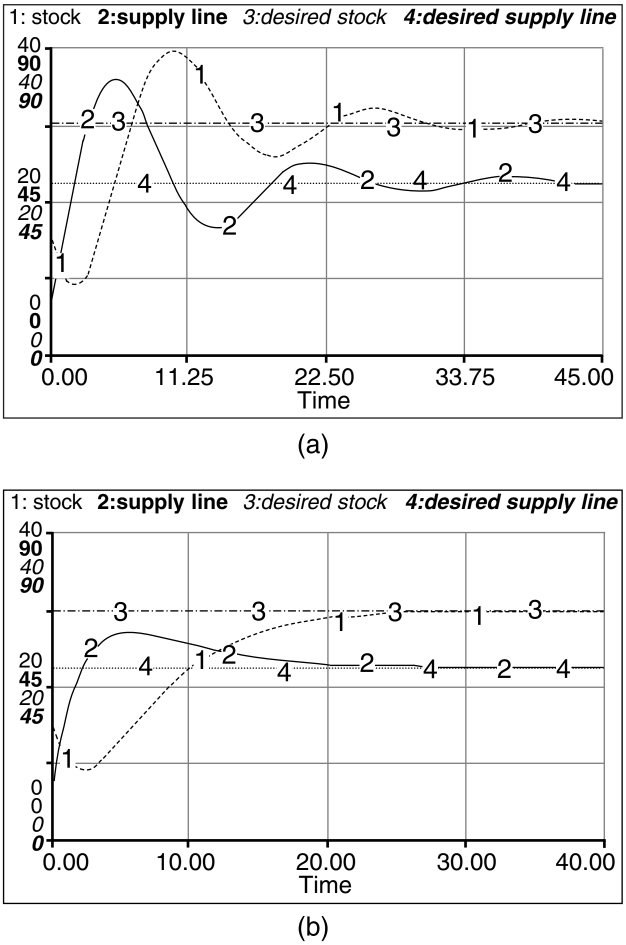


Figure 19. The behavior of the stock adjustment structure: (a) Goods in supply line are ignored in the order decision (b) supply line information is taken into account. All other parameters are the same

and *chaos*. Analysis of such models is not possible by standard mathematical or even numerical methods. There are special numerical and (limited) analytical techniques to analyze complex non-linear dynamics. (See "Non-linear dynamics and chaos," EOLSS on-line, 2002).

6.2. Discrete versus continuous time

System dynamics models can be continuous-time or discrete time. In a continuous model, the stock equations are of the form:

$$Stock(t) = Stock\ (0) + \int_0^t (\sum_j flows)\, dt$$

In simulation, the above equation is approximated by:

$$Stock(t+dt) = Stock(t) + dt * \sum_j flows$$

It is important to note that in the above continuous model "*dt*" is an arbitrarily small computational step. It has *no* real life meaning. On the other hand, in a discrete model, the stock equations are:

$$Stock(t+1) = Stock(t) + \sum_j flows$$

So that *dt* is exactly one time step that corresponds to the real discrete time period in the real system. (Such as a month, a decade, a century, a second or five days). Dynamics of discrete systems can be very different than continuous systems. For instance, 1–stock discrete systems can oscillate, but 1–stock continuous systems cannot. First order discrete system can even generate chaotic dynamics. (See "Non-linear dynamics and chaos," EOLSS on-line, 2002). If the model is continuous, care must be exercised in choosing proper *dt* value, because a large *dt* value may produce spurious dynamic behaviors as a result of numerical solution errors.

6.3. Numerical integration and software

Proper selection of *dt* is not the only issue in obtaining correct simulated behaviors. There are numerous numerical integration methods, ranging from simple and fast ones to very sophisticated yet slow ones. The simplest is Euler's method, which is also used as the default form of stock equation:

$$(Stock\ (t+dt) = Stock\ (t) + dt^{*}) \sum_j flows).$$

There are higher order methods such as high order Taylor methods, Runge-Kutta methods and various adaptive, self-correcting methods. For some methods (like Euler's) small enough *dt* selection is crucial, while other methods vary their own *dt* adaptively as needed, during simulation. (See "Simulation software and numerical considerations," EOLSS on-line, 2002). Finally, numerical analysis of non-linear structures producing complex non-linear dynamics (such as bifurcations, period-doubling, chaos) necessitates special numerical procedures. (see "Non-linear dynamics and chaos," EOLSS on-line, 2002)

Several user-friendly modeling and simulation software exist for system dynamics. The most popular ones are STELLA, Vensim, Powersim and DYNAMO. These types of software are all equipped with different numerical simulation procedures, as well as some other modeling and analysis support tools. (see "Simulation software and numerical considerations," EOLSS on-line, 2002).

6.4. Noise (randomness) in models

System dynamics models can be deterministic as well as stochastic. Randomness can be included in parameters (like delay constant or productivity coefficient) or external input variables (such as demand or stock prices). In any case, the common practice is to include the randomness in the model late in the process, after the model has been completed and tested. Dynamics of a systemic feedback model are primarily created by the structure of the model. Premature inclusion of noise in model parameters will unnecessarily complicate the construction, testing and analysis of the model. After the model is fully tested, the purpose of including noise in selected parameters and variables is to see if there are some "dormant" modes of behaviors (such as oscillations at different frequencies) that would reveal themselves

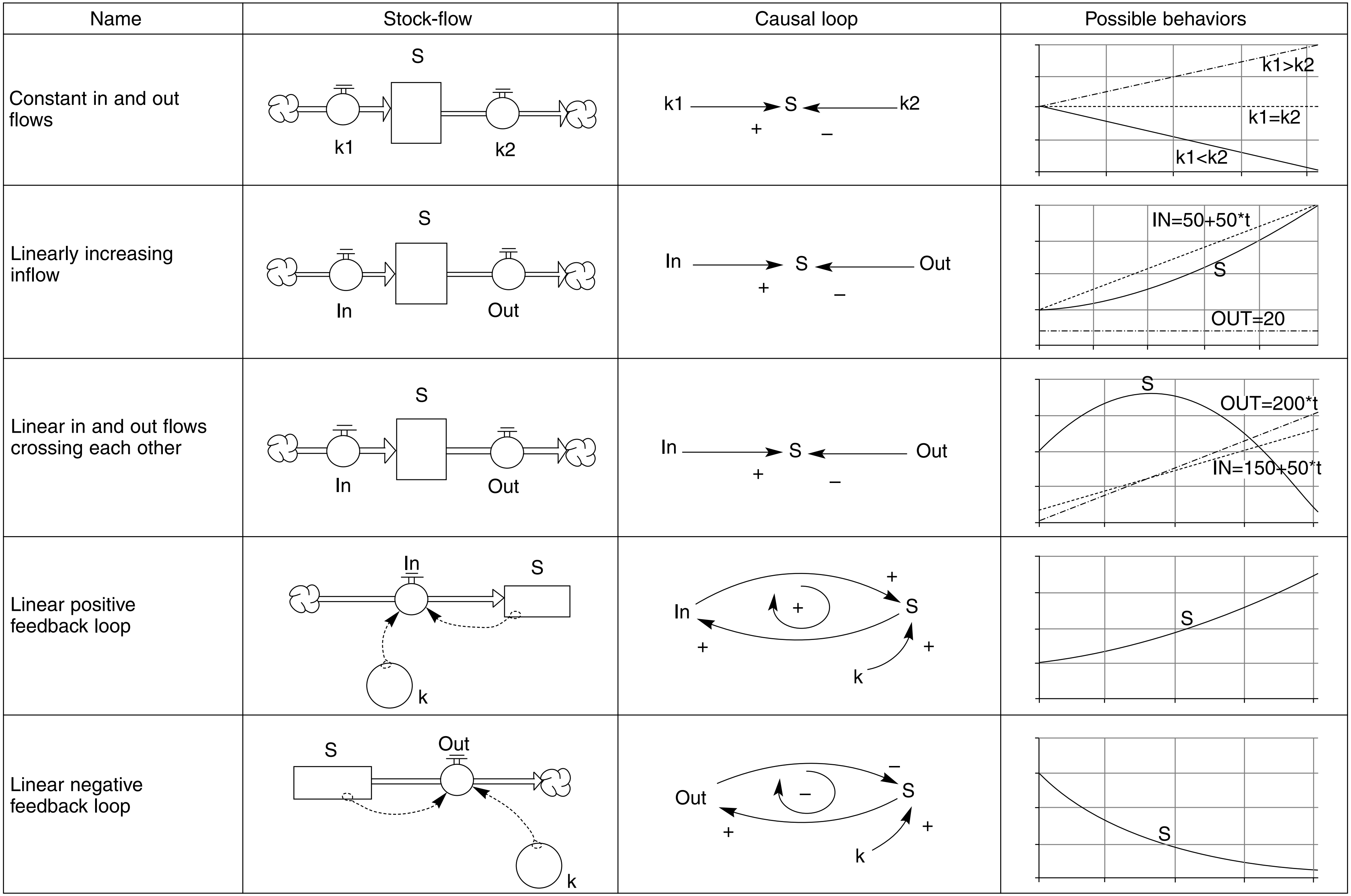


Figure 20.1. Basic generic structures, causal loop diagrams and possible behavior patterns

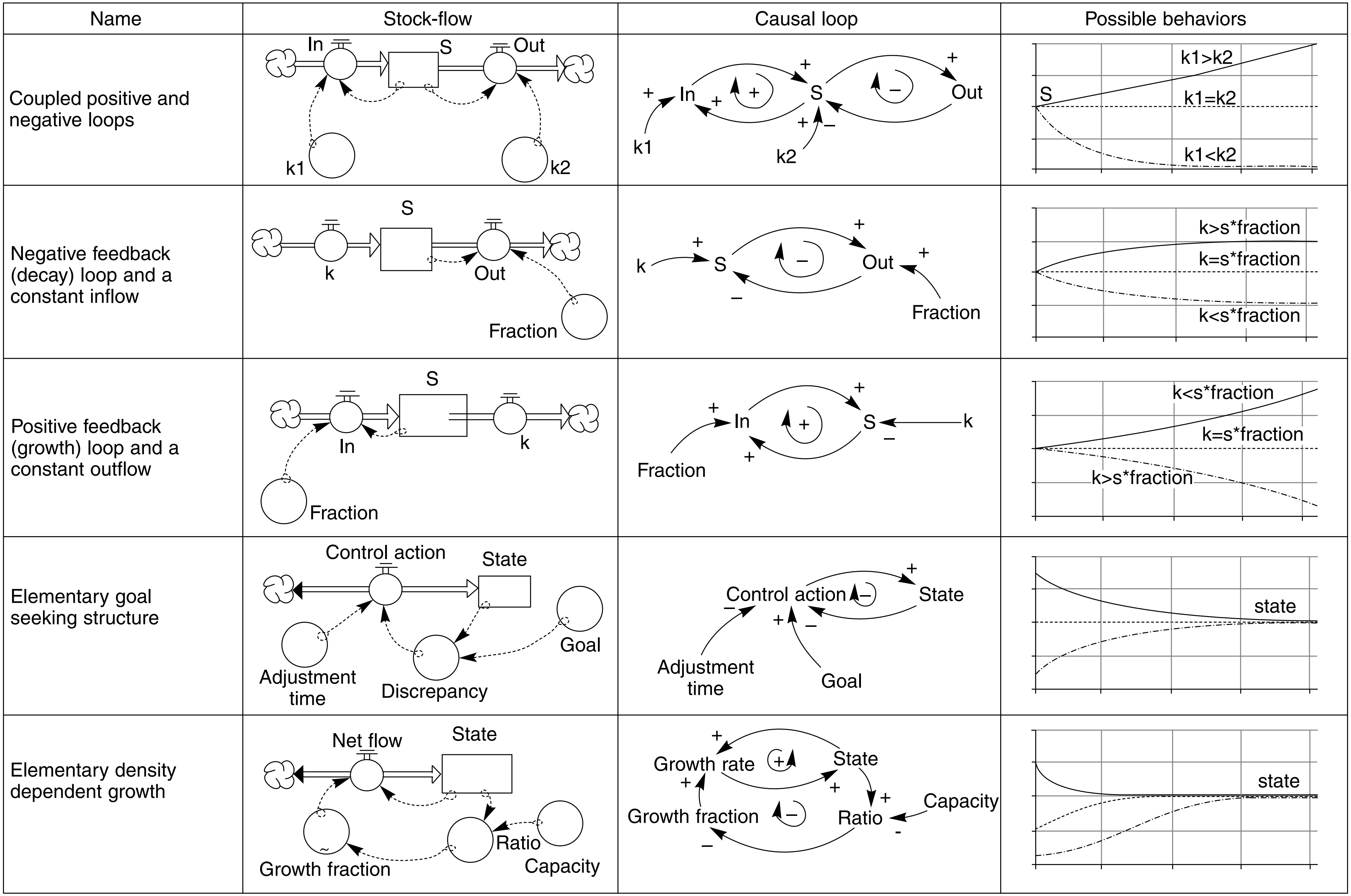


Figure 20.2. Basic generic structures, causal loop diagrams and possible behavior patterns

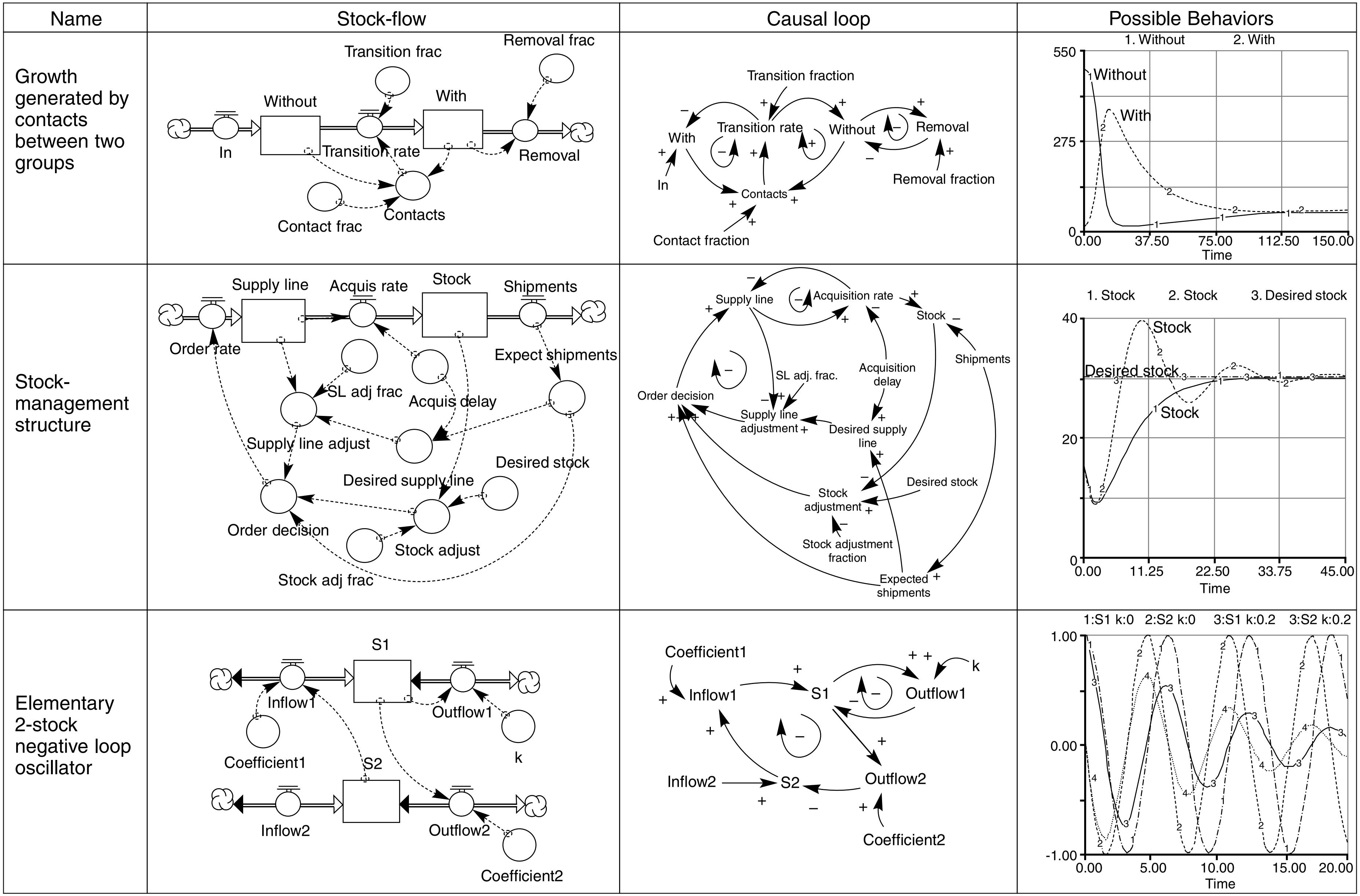


Figure 20.3. Basic generic structures, causal loop diagrams and possible behavior patterns

once excited with resonant noise frequencies. Large-scale feedback models are typically extremely insensitive to independent ("white") noise sequences. Such non-correlated noise sequences are immediately absorbed by the feedback loops of the model so that the dynamic behaviors remain essentially unaltered. In order to be able to excite the model with noisy input, auto correlated noise ("pink" noise) must be used. Auto correlated noise can be obtained by passing white noise through an exponential smoothing filter. In general, the larger the smoothing time, the more auto correlated the noise will become (see Plate 6.63–1 for an illustration). How much autocorrelation is needed in the noise input depends on the structure and natural oscillations of a given model.

7. MODEL TESTING, VALIDITY, ANALYSIS, AND DESIGN

7.1. Model testing and validity

As mentioned earlier in the steps of system dynamics methodology, the model must be tested thoroughly, before policy design and implementation phases. The model must first be tested for internal consistency: Is the simulation model an accurate representation of the conceptual model (the dynamic hypothesis)? Does the model do what the modeler intends it to do? This question is known as the model "verification" problem in simulation. The purpose is to assure that there are no inconsistencies between the model and the dynamic hypothesis. Verification tests also include checking the dimensional consistency of the model and to make sure that the simulation step *dt* is small enough to yield acceptable numerical accuracy.

Model validity (credibility) testing is the next step. The purpose here is to establish that the model is an acceptable description of the real system with respect to the dynamic problem of interest. Model credibility is established by two types of tests:

- Structure tests: is the structure of the model a meaningful description of the real relations that exist in the problem of interest?
- Behavior tests: are the dynamic patterns generated by the model close enough to the real dynamic patterns of interest? (see "Model testing and validity," EOLSS on-line, 2002).

Examples of structural tests are: having experts evaluate the model structures, and robustness of each equation under extreme conditions and extreme condition simulations. Behavior tests are designed to compare the major *pattern components* in the model behavior with the pattern components in the real behavior. Such pattern measures include slopes, maxima and minima, periods and amplitudes of oscillations (autocorrelation functions or spectral densities), inflection points, and so on. Behavior testing does not involve point-by-point comparison of model behavior with real behavior; it involves comparing of *patterns* involved in the two. It is important to note that unless structural validity is established first, behavior validity is meaningless in system dynamics modeling. (see "Model testing and validity," EOLSS on-line, 2002).

7.2. Model analysis and design

Model analysis can be defined as "understanding why and how the model behaves the way it does." Analysis can be done mathematically when possible, as mentioned above (analytical solution, equilibrium points, stability analysis). Since this is often impossible, analysis is more typically done by *sensitivity analysis*: how and to what extent the behavior of the model changes, as a result of changes in its parameters and/or structures. Sensitivity analysis can be of different types (with respect to structures and with respect to parameter values; numerical sensitivity and behavior pattern sensitivity) and there are various methods to do sensitivity analysis. (see "Sensitivity analysis," EOLSS on-line, 2002).

Policy analysis is about the sensitivity of model behavior to the policy parameters and/or policy structures. Policies represent rules for conscious human control. They may be characterized by:

- set of parameter values (numeric)
- function values (numeric)
- function shapes (structural)
- forms of policy equations (structural)

Policy analysis and design may involve altering one or more of the above model characteristics and examining the resulting behavior. Like sensitivity analysis, policy analysis can also be numerical or pattern-oriented. Pattern-oriented policy analysis is naturally much more important, since the purpose of system dynamics studies is to improve undesirable dynamic behavior patterns. Policy design is determining what changes in the model structure and parameters ("improved policy") would lead to an improved model behavior.

No matter how scientific and exact the policy design phase is, naturally the ultimate success depends on implementation. There are some general guidelines for implementation success that system dynamics researchers have developed. (see "Implementation issues," EOLSS on-line, 2002). There are also new important developments that seek to enhance implementation success, such as group model building (see "Group model building," EOLSS on-line, 2002) and interactive learning environments (see "Modeling for learning and interactive environments," EOLSS on-line, 2002).

8. CONCLUSIONS AND FUTURE

System dynamics discipline emerged in the late 1950s, as an attempt to address dynamic, long-term policy issues, both in the public and corporate domain. Since then, applications have expanded to a very wide spectrum, including national economic problems, supply chains, project management, educational problems, energy systems, sustainable development, politics, psychology, medical sciences, health care, and many other areas. Today, interest in system dynamics and related systemic disciplines is growing very fast. At present there is a bottleneck in formal education for system dynamics expertise. As Jay W. Forrester puts it: "new fields, like system dynamics, that cut across boundaries of existing fields, but do not lie within any one of those fields, lack a home and a supporting constituency in universities." A major challenge for the next decade or so is to design and set up educational institutions that can meet the growing need for system dynamics and systems thinking education. These institutions should cover the entire spectrum; from K–12 to post graduate education. There is already evidence that great success can be obtained in K–12 education by combining system dynamics with "learner-directed" learning. (see "On the history, the present and the future of system dynamics," EOLSS on-line, 2002).

The second major challenge for system dynamics is to overcome the "implementation" hurdles. Traditional system dynamics studies have been in the "external consultant" mode, in which a modeling expert (consultant) studies a problem and comes up with recommendations. Unfortunately, the implementation success record in using this approach has been rather slim. There are numerous important causes of the implementation failure beyond the scope of this section. (see "Intellectual roots and philosophy of system dynamics," and "Implementation issues," EOLSS on-line, 2002). But they can perhaps be summarized as: "the collective mindset, the jargon of the real-system participants and that of the external consultant do not match." The real system therefore cannot internalize the design recommendations made by the external expert. To overcome this hurdle, there have been some significant developments in the areas of "group model building" and "modeling for learning." In "group model building," system participants take part in all the phases of the system dynamics project. The idea is to make full use of the "mental models" of the system participants in building the system dynamics model, as well as to give them time and opportunity to modify their mental models so that internalization of model recommendations is possible. (see "Group model building," EOLSS on-line, 2002). In the "modeling for learning" approach, models are typically small, they are built in a group process (including non-technical participants) and their primary use is to enhance (organizational) learning. Another use of system dynamics modeling that often supports "modeling for learning" is interactive simulation gaming ("management flight simulators"). In this mode, players make some decisions interactively during the course of the simulation. Such interactive games can be used in setting up "organizational learning laboratories." (see "Modeling for learning and interactive learning environments," EOLSS on-line, 2002).

System dynamics has the potential to make revolutionary contributions to education and learning in general. In management education, the traditional "case study approach" is already being complemented by system dynamics models ("model-based case studies"). In model-based case studies, students can experiment on simulation models of the case, which gives them a chance to rigorously test their own theories of how the problems in the case could be avoided. A similar approach can be used in various disciplines ranging from psychology to political science, from physics to medicine, from environmental science to economics.

System dynamics has a unique set of tools to offer as we enter the twenty-first century. It can address the fundamental structural causes of the long-term dynamic contemporary socio-economic problems. Its "systems" perspective challenges the barriers that separate scientific disciplines. The problems in the real world have no "discipline barriers," they have social, economic, psychological, political, and technical aspects, all interacting in a complex manner. Many education experts believe that the current heavily compartmentalized education will change into a more integrated education in the 21st Century. The interdisciplinary and systemic approach of system dynamics could be critical in dealing with the increasingly complex problems of our modern world: from hunger to ecological problems; from globalization to unemployment, from wars to education.

KNOWLEDGE IN DEPTH

In depth knowledge of this subject is available in several chapters in the on-line *Encyclopedia of Life Support Systems*, organized as follows:

Systems Dynamics: Systemic Feedback Modeling for Policy Analysis

1. Systems Dynamics in Action: Selected Examples

Urban dynamics, *Louis Edward Alfeld, USA*

Supply-chain dynamics, the "Beer Distribution Game" and misperceptions in dynamic decision-making, *John Sterman, Sloan School, MIT, Cambridge, Mass., USA*

Market growth, collapse and failures to learn from interactive simulation games, *John Sterman, Sloan School, MIT, Cambridge, Mass., USA*

A dynamic model of cocaine prevalence, *Jack B. Homer, Homer Consulting, USA*

Ecological interactions: predator and prey dynamics on the Kaibab plateau, *Andrew Ford, Program in Environmental Science and Regional Planning, Washington State University, USA*

R&D, technological innovations and diffusion, *Peter M. Milling and Frank H. Maier, Industrieseseminar der Universitat Mannheim, Mannheim University, Germany.*

2. Conceptual and Philosophical Foundations*, Yaman Barlas, Department of Industrial Engineering, Boğaziçi University, Turkey*

On the history, the present and the future of system dynamics, *Jay W. Forrester, MIT System Dynamics Group, Sloan School of Management, Massachusetts Institute of Technology, USA*

Different philosophical positions and system dynamics, *Yaman Barlas, Department of Industrial Engineering, Boğaziçi University, Turkey*

GLOSSARY

Dynamic behavior: dynamic performance patterns of the variables created by the operation of the structure of the system over time.

Dynamic problems: problems characterized by variables that undergo significant changes in time. Being chronic in nature, they necessitate continuous managerial monitoring, control, and action.

Endogenous: the dynamics are essentially caused by the internal feedback structure of the system.

Feedback: a succession of cause and effect relations that starts and ends with the same variable. Also called "loop" or "circular" causality.

Flows: They directly flow in and out of the stocks, thus changing their values. They represent the "rate of change" of stocks.

Generic structures: fundamental compact model structures frequently used as basic building blocks in diverse applications.

Model: a representation of selected aspects of a real system with respect to some specific problem(s).

Negative loop: a feedback loop that balances, counteracts an initial change in any of its variables. It produces a goal-seeking behavior.

Non-linearity: any relationship where the output is not purely proportional to the input; any relationship that is not linear.

Positive loop: a feedback loop that reinforces, compounds an initial change in any of its variables. It produces an exponentially growing or collapsing behavior.

Simulation: a step-by-step operation of the model structure over compressed time; imitation of the operation of the real system.

Stocks: They represent the important accumulations over time. They are also called "states" as they collectively represent the state of the system at time *t*.

Structure: the totality of the relationships that exist between system variables.

Systemic: resulting from the complicated interactions between many variables in the system.

Time delay: a time duration that intervenes before a cause can reach its effect. Delay can exist in physical flows or in informational cause-effect relations.

BIBLIOGRAPHY

BARLAS, Y. 1996. Formal Aspects of Model Validity and Validation in System Dynamics. *System Dynamics Review,* Vol.12, No.3, pp. 183–210. [A comprehensive overview of model validity and validity testing in system dynamics, including philosophical foundations as well as specific tests.]

COYLE, R G 1996, *System Dynamics Modelling: A Practical Approach*, London, Chapman and Hall, now available from CRC Press. [A treatment of both qualitative and quantitative system dynamics modeling, as well as the optimization of models.]

FORD, A. 1999. *Modeling the Environment*, Island Press. [An introductory textbook on system dynamics, as well as environmental and energy applications.]

FORRESTER, J. W. 1961. *Industrial Dynamics*. Waltham, Mass., Pegasus Communications. 464 pp. [The book establishes the foundation of system dynamics as a new discipline. It contains conceptual foundations, several business applications, and very useful technical appendices.]

———.1969. *Urban Dynamics*, Waltham, Mass., Pegasus Communications. 285 pp. [Presents a classical model of urban growth, stagnation, and possible revival. Simulations show why popular policies for improving cities can fail.]

———.1975. *Collected Papers of Jay W. Forrester*. Waltham, Mass., Pegasus Communications. 284 pp.[Seventeen classical papers on management, corporate organization, urban policy, and the global environment.]

GUTIERREZ, L. T.; FEY, W. R. 1980. *Ecosystem Succession*. Cambridge, Mass., The MIT Press. [This is one of the most extensive system dynamics analysis of a large-scale, non-human living system (grassland biome). It illustrates the important variables and feedback loops that create most biomes' dynamic patterns and the philosophy of studying non-human systems.]

MEADOWS, D. H.; MEADOWS, D. L.; RANDERS, J.; BEHRENS III. W. W. 1972. *The Limits to Growth*. New York, Universe Books. 205 pp. [A popular book on world dynamics that has been translated into many languages, and has sold millions of copies.]

MORECROFT, J. D. W.; STERMAN, J. D. (eds.). 1994. *Modeling for learning organizations*. Portland, Ore., Productivity Press. [This book contains a collection of papers discussing the rationale behind modeling for learning, issues like knowledge elicitation, and examples of group model building in action.]

RICHARDSON, G. P. 1999. *Feedback Thought in Social Science and Systems Theory*. Waltham, MA., Pegasus Communications. [A comprehensive text that traces the history of feedback thinking in the social sciences and systems theory.]

ROBERTS, N, ET.AL. 1983. *Introduction to Computer Simulation: A System Dynamics Approach,* Portland, Ore., Productivity Press. [A very useful introductory book on systems thinking and system dynamics, especially for high school and non-technical university students.]

SAEED, K. 1998. *Towards Sustainable Development, 2nd Edition: Essays on System Analysis of National Policy*. Aldershot, UK, Ashgate Publishing Company. [Readings on system dynamics applications to sustainable developmental agendas.]

SENGE, P. M. 1990. *The fifth discipline. The art and practice of the learning organization*. New York, Doubleday/Currency. [A very popular text that introduces system dynamics and systems thinking to the business world, placing them in the context of organizational learning, together with sister disciplines.]

STERMAN, J. D. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. New York, Irwin. McGraw-Hill. 982 pp. [An excellent, very comprehensive text book for the study of system dynamics.]

System Dynamics Review. 1985. Chichester, UK, John Wiley. [Main scientific journal for system dynamics, publishing methodological articles as well as a wide range of applications.]

VENNIX, J. A. M. 1996. *Group model-building: facilitating team learning using system dynamics*. Chichester, UK, John Wiley. [This book focuses on Group Model Building, including theoretical/conceptual chapters as well as real application projects.]

ACKNOWLEDGMENTS

I am grateful to my student-assistants Özge Karanfil and Özge Özdemir for their meticulous help in preparing the figures.

BIOGRAPHICAL SKETCH

Yaman Barlas received his B.S., M.S. and Ph.D. degrees in Industrial and Systems Engineering, from Middle East Technical University, Ohio University and Georgia Institute of Technology respectively. In 1985, upon receiving his doctoral degree, he joined Miami University of Ohio as an assistant professor of Systems Analysis where he was granted tenure in 1990. He worked as a guest researcher at MIT in the summer of 1990. In 1992, he gave seminars in Istanbul, as a United Nations TOKTEN consultant. He joined Boğaziçi University in 1993, where he is currently working as a professor of Industrial Engineering and directing the SESDYN research laboratory (http://www.ie.boun.edu.tr/labs/sesdyn/). He spent his sabbatical leave at University of Bergen in Norway in 2000.

His interest areas are validation of simulation models, system dynamics methodology, modeling/analysis of socio-economic problems and simulation as a learning/training platform. He has published numerous articles in journals, proceedings and books, and offered academic and professional seminars nationally and internationally in these areas. He teaches simulation, system dynamics, dynamics of socio-economic systems and advanced dynamic systems modeling. He is a founding member of the System Dynamics Society and member of several other international and national professional organizations. Professor Barlas was the Chair of the 15th International System Dynamics Conference, held in Istanbul in 1997. He is the editor of *System Dynamics Review* for short articles, an invited Honorary Editor of the Encyclopedia of Life Support Systems and a former President of the System Dynamics Society.