Theory of Complex Work

Larsen, Harry

Unaffiliated

10 June 2022

Theory of Complex Work

Harry Larsen
The Boeing Co. (Retired) harrylarsen@comcast.net

Abstract

Work in general, and the learning curve in particular, are typically modeled with the power law, **$y(x) = a\ x^b$**. This model provides little insight into the causes of the dynamics of the labor hours associated with work. A stochastic model with links to information is proposed. It is implemented as a Monte Carlo system, producing probability density functions, thus deriving labor cost uncertainty as intrinsic to the performance of work itself. The paper demonstrates a time domain application of the model in a factory setting with feedback interactions between work elements.

Keywords: Learning curve, Slope, Negative Binomial Distribution, Task, Trial, Uncertainty

Introduction

The manufacturing learning curve graph depicts the direct labor hours per unit of production plotted in its manufacturing sequence. When plotted with log/log scales, learning curves tend to follow a linear downward trend as shown in figure 1.

The power law, **$y(x) = a\ x^b$**, has been found to describe the learning curve's trend, with **$y(x)$** = hours(unit), **x** = sequential unit number, **a** = the hours at unit one and **b** the slope exponent. The power law is transformed to a linear form by taking its logarithm:

$\ln(\mathbf{y}) = \ln(\mathbf{a}) + \mathbf{b}\ln(\mathbf{x})$. To summarize a learning curve's statistics, a least squares regression can be performed on the logarithms of learning curve data, $(\ln(y_1), \ln(1))$; $(\ln(y_2), \ln(2))$; $\cdot\cdot\cdot$, $(\ln(y_n), \ln(n))$. The regression's estimate of **a** is termed the theoretical number one. By convention the slope of a learning curve is $2^{\mathbf{b}}$.

The learning curve is sometimes transformed into a cumulative average:

$$y_{cum}(x) = \sum_{i=1}^{x} y(i)/x$$

For this discussion, the hours per unit description will be used.

From an examination of the historic learning curve labor hours in the airframe industry, some attributes are apparent.

The slope of a long sequence of end item's labor hours is very seldom steeper than 70.7%, **b**= -.5.

When new work, e.g., a new design, for part of an end item is introduced at a unit, the hours for the new work are approximated by **$y(x) = a_{new}\ x^b$**, with **x** beginning at unit 1. The replaced work is removed via **$y(x) = a_{replace}\ x^b$**, with **x** at its current value. With **a_{new}** = #1 hours for the new work and **$a_{replace}$** = the #1 hours for the replaced work. This produces a spike in the learning curve.

Processes earlier in the manufacturing process have flatter slopes than do the later activities.
In the aircraft industry major and final assembly curves often steepen as they progress, while fabrication curves typically are less steep.
The learning curves of very large quantities of end items flatten.
The learning curves of products with large aggregates of hours per unit tend to be smoother than learning curves of fewer hours per unit. Component learning curves of an end item have a larger variation in hours between units than do their sum.
The probability distributions about the power law regression of labor hour data are approximately log-normal. (This is true for both learning curve hours per unit as well as Cost Estimating Relationships (CERs) for labor hours.)
There is a large cost variation around estimates for complex projects, particularly those that involve new technology.
For large complex products, e.g., a commercial airplane, the hours at unit 1000+ may be as little as .10 to .03 of the hours of the first unit.

Although the power law describes broadly the learning curve's log-linear behavior, it does not offer an insight into the causes or mechanisms of its realization. Nor does it suggest means by which its labor hours per unit may be affected.
Learning curves are a unit-by-unit compilation of labor hours, as such, they constitute a series. Their values, transformed to a stationary series, typically have strong autocorrelations. The power function coupled with the linear regression does not model this relationship. Consequently, it does not lend itself to the projection of future values of existing learning curves. A Markov chain is proposed to model the learning curve.

The theory

This model of work is based on the concept of a task.
The construction of a product is accomplished by the successful completion of a set of tasks.
A task is an activity that has a criterion of success.
A task is accomplished through repetitive trials.
The trials are repeated until the criterion of success is met.
A trial is modeled as a binomial event, with probability **p(t)**, and **t =** trial.
After each trial, the probability of success, **p(t),** is increased by an amount **(1 - p(t)) dl,** where **dl** is a constant << **p(0)** and approximately equal to **p(0)**^{2}.

$$p(t+1) = p(t) + (1 - p(t))\, dl$$

The duration of each trial is constant.
Thus, a task's labor hours are proportional to the number of trials necessary to complete the task successfully. A task may be a part of a larger interrelated group of tasks. The outcome of a task's successful trial may cause another task's success criterion to change, requiring it to be redone.
The successful completion of a product's tasks produces a unit of the learning curve.

To model this process at each trial **p(t)** is compared to **x ~** U(0,1), an event from a uniform distribution. If **x** < **p(t)** the trial is a success and the task for that unit is completed, otherwise the trial is repeated. Trials for the next unit's task begin with the **p(t)** from the preceding completed task. The sequential completion of tasks produces the learning curve.

This is a stochastic process, with inherent uncertainty. To produce an estimate, it is implemented as a Monte Carlo system, producing Probability Density Functions, (PDFs). From those PDFs various statistics can be calculated, e.g., median, mean, and standard deviation. Or, if the estimate is to support a decision option, the PDF can be bifurcated.

Fig 1 depicts a learning curve created by such a sequential completion of tasks. Its initial probability, **p(0)**, is .03 while **dl** equals .001, with 400 tasks and a trial time of 1 hour.

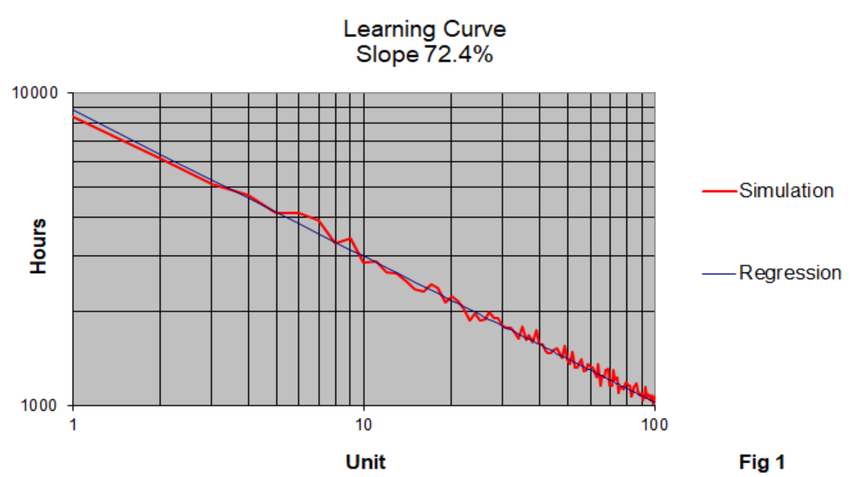


Fig 1

Predictions

The expected value of the number of trials to successful completion is approximately 1/Mean(**p(t)**). Thus, with **p(t)** small, the change in **p** between successful trials is about **dl/**Mean(**p(t)**). If this value is small the number of trials to successfully complete a set of tasks is a negative binomial distribution. [1]

The mean of a negative binomial distribution is:

(1) $\mathbf{u = T\ N\ /\ p}$, where **N** is the number of tasks, **T** is the time per trial for a task.

its standard deviation,

(2) $\boldsymbol{\sigma = T\ (N\ (1\text{-}p))^{.5}\ /\ p,}$

The relative standard deviation is

$\boldsymbol{\sigma /\ u = (T\ (N\ (1\text{-}p))^{.5}\ /\ p)\ /\ (T\ N\ /\ p)}$

---

[1] A negative binomial distribution can be defined as a discrete probability distribution of the count of failed trials up to a successful trial in a set of experiments. For this application, the count includes the successful trial.

or

(3) $\sigma/u = ((1-p) / N)^{.5}$

The number of tasks is: From (3)

(4) $N = (1-p) / (\sigma/u)^2$

From equation (1) it can be seen that a project's expected hours are proportional to the size of the project in terms of the number of tasks, **N**, trial duration, **T**, and the difficulty of the project measured by the inverse of a trial's probability of success, **p**. Thus, a project may be large in labor hours either due to its task content or its difficulty.

From equation (2), for **p** << 1, the standard deviation of labor hours is proportional to the square root of the number of tasks and the inverse of a trial's probability of success. Consequently, as illustrated by the relative standard deviation formula (3), a project with a higher task content, while holding **p** constant, will have a relatively lower standard deviation. Hence its depiction on log/log scales becomes smoother as the project's task content increases. Conversely, for a project that increases in size due to increased difficulty (smaller **p**), its sequential standard deviation remains directly proportional to the increased labor hours and does not collapse with the larger project size. Thus, we should expect that large projects, that advance the state of the art, will have high relative standard deviations, and projects that do not, will have low relative standard deviations.

For small **p** the relative standard deviation, **σ/u,** is a function of **N**. Figure 2 is a graph of a learning curve with 200 tasks and 40 tasks.

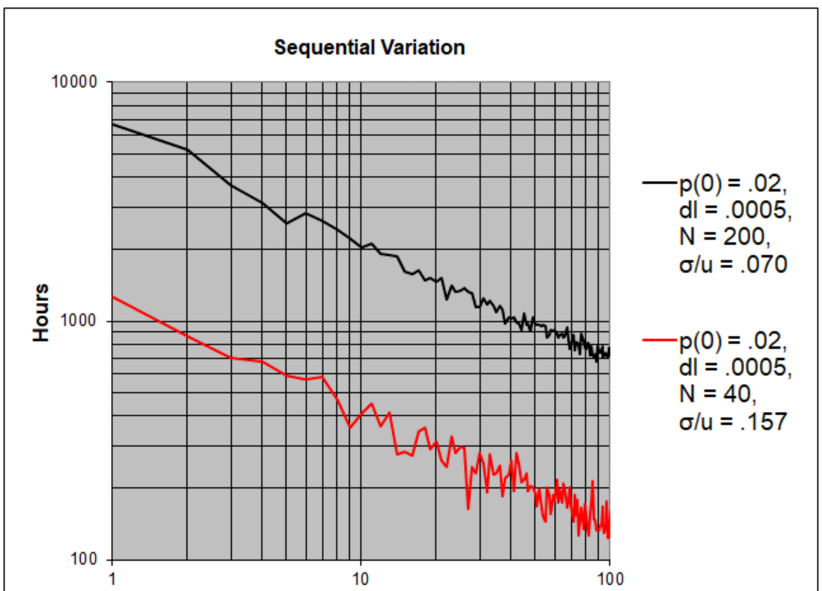


Fig 2

For very large quantities a learning curve may flatten. When **p** approaches 1 the hours per unit approach the product of the number of tasks times and the hours per trial, as shown in figure 3.

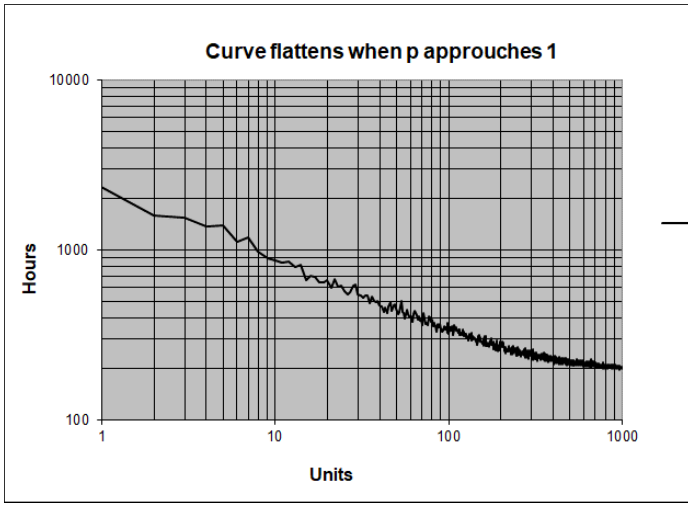


Fig. 3

Figure 4 illustrates that **p(0)** and **dl** determine the hours for the first unit. As **p(0)** is increased the learning curve exhibits a lower initial cost but remains asymptotic to a slope of 70.7% until **p** approaches 1.

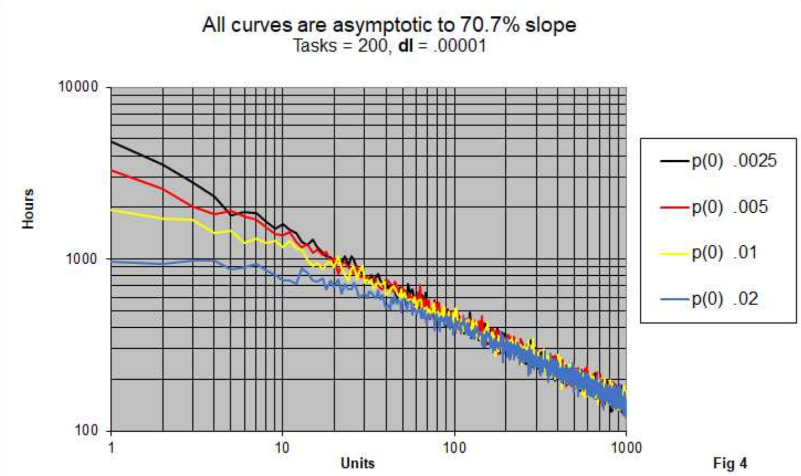


Fig 4

A combination of learning curves with differing **p(0)** values can produce a curve with a shallower slope, as shown in figure 5.

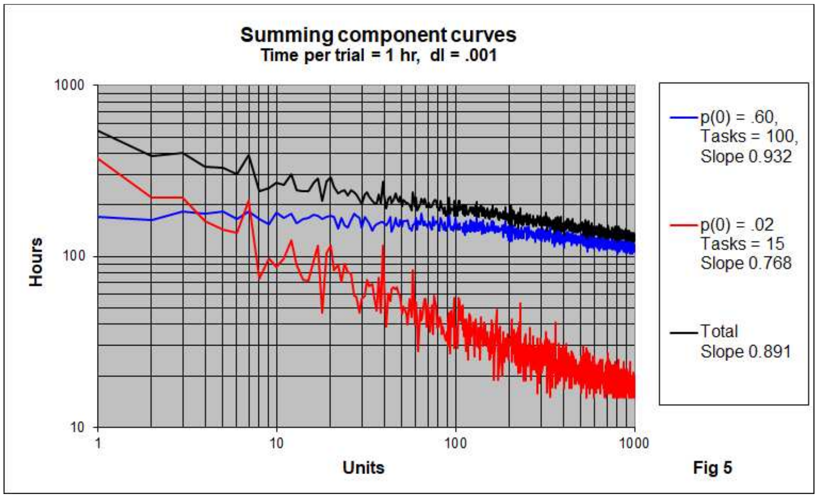


Fig 5

More realistically, a distribution of **p(0)** values can produce a learning curve with a small hump followed by a flattening. Figure 6 is a histogram of a lognormal distribution of 200 **p(0)** values, with a mean of .233 and a standard deviation of .133. Figure 7 is the resulting learning curve of 200 corresponding tasks beginning with those p(0) values. A log-log regression through unit 500 of the 1000 units shown has a slope of .822, a slope in the range commonly seen in aircraft learning curve history.

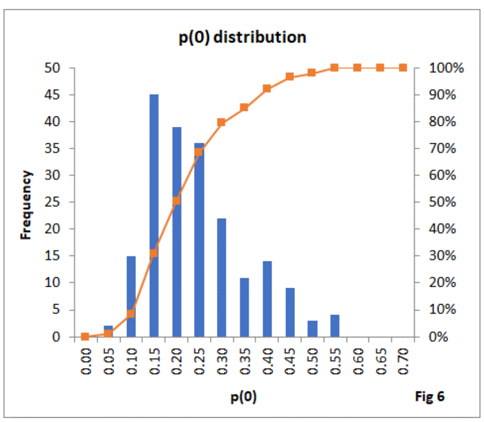


Fig 6

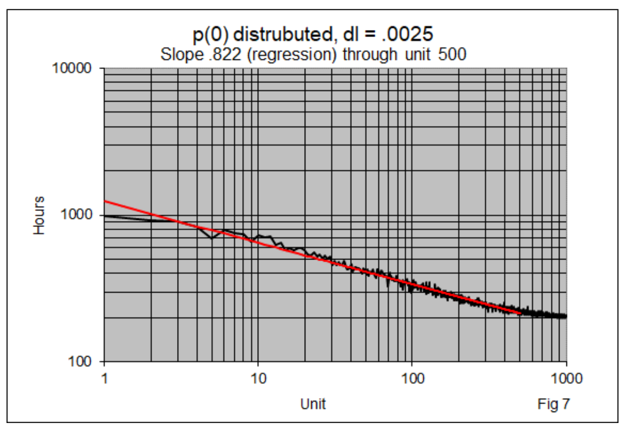


Fig 7

If a design change is introduced to an existing production process, it is added at the initial probability of the task, **p(0)**. The work replaced is removed at its current probability, **p(t).**

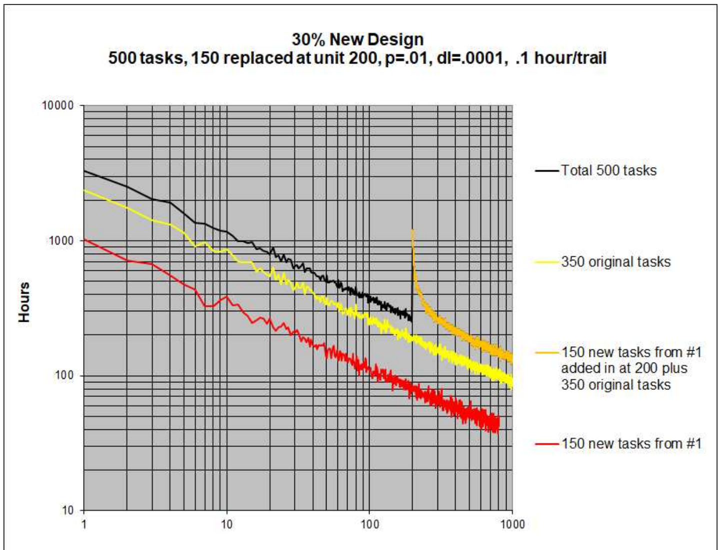
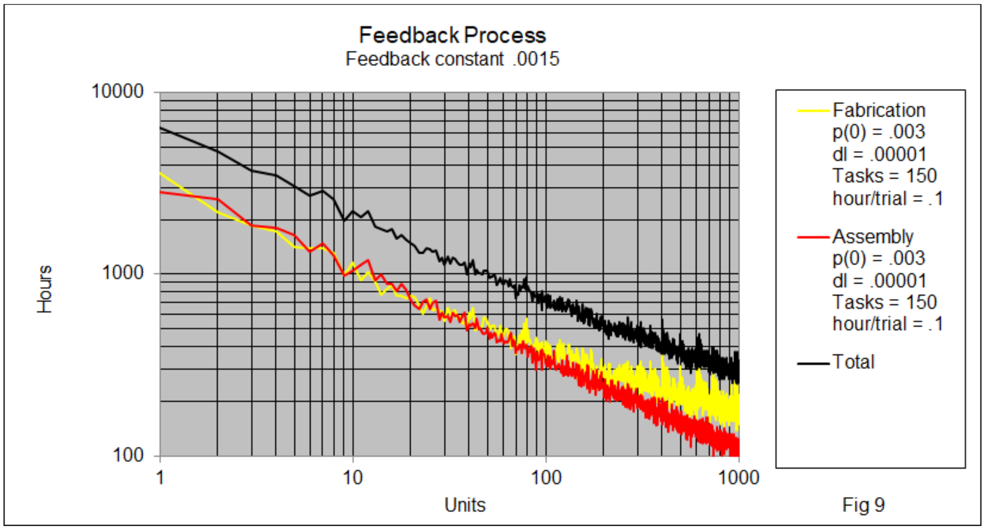


Fig 8

In figure 8 the yellow line represents 70% of the tasks, those that are unchanged. The red line scaled from one shows the new tasks. The orange, beginning at unit 200, shows the sum of new tasks and the continued unchanged tasks.

Feedback may occur between its tasks. When, for example, in an assembly task it is found that a part does not fit due to a design or manufacturing error, the part may be reworked in the assembly activity to fit, but also, a design or specification change may be fed back to the task that built the part. The design change is treated as a new design, setting that task's current p(t) to p(0) as shown in figure 8, but for a single task. Figure 9 shows a simulation of two sets of 150 tasks, fabrication and assembly. Design changes are fed back from assembly to fabrication. The likelihood of a fabrication task's **p** being reset to **p(0)** is proportional to the product of .0015 and the ratio of the number of trials for the preceding unit in assembly versus those in fabrication. Without feedback, the fabrication curve would follow the trend of the assembly curve.

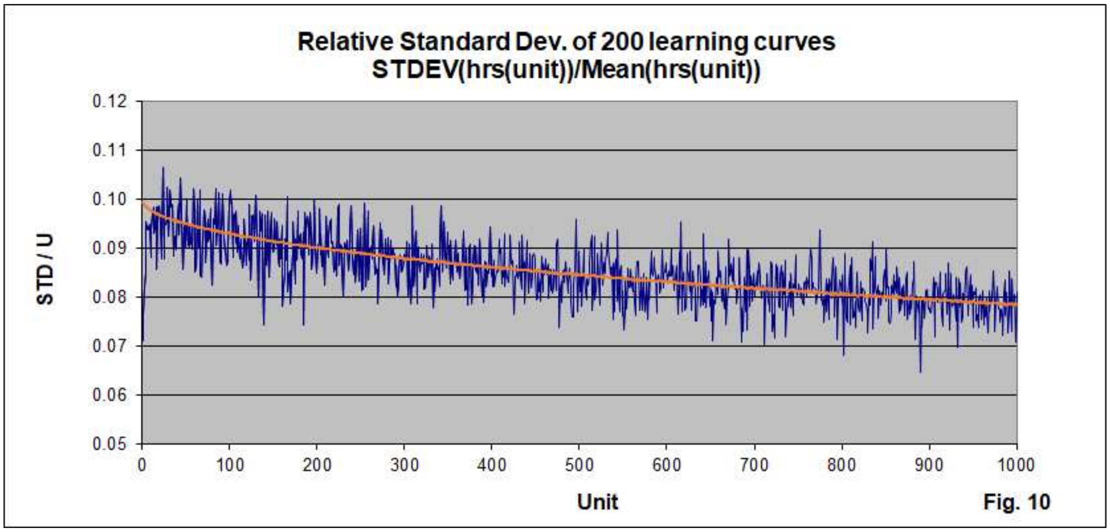


The derivation of model parameters

If **p** is small, from eq. (4) **N** can be calculated, $\mathbf{N = (1\text{-}p) / (\sigma/u)^2}$. To calculate $\boldsymbol{\sigma/u}$, a learning curve must be transformed into a stationary series. One approach to doing so is to divide the curve by its unit number taken to its slope exponent: $\mathbf{y(x)} = \mathbf{hours(x) / x^{(slope\ exp)}}$. If **y(x)** is autocorrelated a time series model can be used to remove the autocorrelations. Then the standard deviation of the sequential difference of the series: **STD(var)** with **var** = **y(x+1)** - **y(x)** can be calculated. (It can be shown that sequential differencing increases the **STD** by $\mathbf{2^{.5}}$. Thus $\boldsymbol{\sigma/u}$ = (**STD(var)** / $\mathbf{2^{.5}}$ **) / mean(y(x))**). From eq. (1), **u = T N / p,** thus **p = T N /u** and the average **p** for unit 1, $\mathbf{p}(\text{unit 1})_{\mathbf{mean}}$, equals **T N /u**(unit1)**.** For a learning curve of 70.7% slope, **p(0) ~ .58 p**$(\text{unit 1})_{\mathbf{mean}}$. With a choice of **p(0)**, **dl** for a straight learning curve with a 70.7% slope is **~ 1.6** $\mathbf{p(0)^2}$. Since a task is defined as a set of work that has a criterion of success, **T** may be evaluated through an industrial engineering study.

This paper contains no actual labor hour data. Although it is common for manufacturing companies to track labor hours per unit and occasionally it is published, none has been found available for inclusion in this paper. The US DoD has compiled aircraft unit labor hours since before WWII as have some large aircraft companies, but its availability is restricted.

The equation for the negative binomial distribution's relative standard deviation, $\boldsymbol{\sigma/u = ((1\text{-}p) / N)^{.5}}$, predicts that $\boldsymbol{\sigma/u}$ will decline as **p** increases. Figure 10 is a graph of the relative standard deviation, $\boldsymbol{\sigma/u}$, of 200 simulated learning curves. Also shown is the $\boldsymbol{\sigma/u}$ from the negative binomial distribution. Thus, it is seen that iteration of the equation, **p(t+1) = p(t) + (1 - p(t)) dl,** closely approximates the $\boldsymbol{\sigma/u}$ prediction of the negative binomial distribution.



Time domain formulation

During the design of a product, there may be information flows between engineering tasks. But unlike a production line, these new requirements are applied to the design tasks themselves rather than to a subsequent design. Modeling the equation,
p(t+1) = p(t) + (1 - p(t)) dl, in time for both engineering and manufacturing allows these information flows to interact within and between the design tasks and factory tasks.
To show these effects a small airplane program is modeled. It produces 300 aircraft over 7 years. Its design effort begins 3 years before the first factory complete aircraft. There are 805 engineering tasks in this simulation each with an associated task in manufacturing.
The engineering tasks are organized into a Work Breakdown Structure, (WBS), of 5 elements, and the corresponding manufacturing tasks into three cost elements, as shown in table 1. In this example, there are 250 Structure design tasks. One hundred twenty-five of those designs are built in fabrication.

Table 1. Work Breakdown Structure by manufacturing cost element

WBS/CE	Structure	Subsystems	Avionics	Systems Eng	Test	Total
Fabrication	125	23	15	8	10	181
Minor Assy	100	69	45	15	20	249
Major Assy	25	138	90	52	70	375
Total	250	230	150	75	100	805

With modeling in time, design problems encounter in minor assembly and major assembly can be fed back to engineering where the design is changed and then sent back to manufacturing. This creates a sustaining engineering effort as well as an increase in the fabrication and minor assembly hours. In this example, there is a .0006 probability, as shown in Table 2, that a trial in a major assembly or minor assembly task will cause a design/fabrication task to be redone, with an additional .0006 that a major assembly trial will cause a design/minor assembly task to be redone. This generates design changes proportional to the hours worked in manufacturing. After the design change is completed it is sent back to its corresponding manufacturing element where its **p** is set to **p(0)**.

Also, within engineering, .05 of Subsystems completed tasks produce a design change in Structure, while .15 of Systems Eng produce design changes in Avionics, and .20 of Test produce design changes in Test. These changes are treated like repetitive units in manufacturing, that is, the probability, **p**, is iterated from its last successful completion.

Table 2. Feedback parameters:

WBS	Structure	Subsystems	Avionics	Systems Eng	Test	Cost			
Structure	0.05	0.05	0.05	0.05	0.05	Element	Fabrication	Minor Assy	Major Assy
Subsystems	0.05	0.05	0.10	0.15	0.20	Fabrication	0	0.0006	0.0006
Avionics	0.05	0.05	0.10	0.15	0.20	Minor Assy		0	0.0006
Systems Eng	0.05	0.05	0.10	0.15	0.20	Major Assy			0
Test	0.05	0.05	0.10	0.15	0.20				

To complete the definition of the simulated airplane program table 3 and figure 11 have the schedule and work parameters.

Table 3. Program parameters

WBS	Structure	Subsystems	Avionics	Systems Eng	Test	Cost Element	Fabrication	Minor Assy	Major Assy
Start	10/01/23	10/01/23	10/01/23	06/01/23	12/01/24	#1 flow months	21	21	21
Planned complete	12/01/25	12/01/25	12/01/25	06/01/25	09/01/26	Flow red. slope	0.94	0.94	0.94
p(0)	0.003	0.003	0.003	0.003	0.003	p(0)	0.003	0.003	0.003
dl	0.00001	0.00001	0.00001	0.00001	0.00001	dl	0.00001	0.00001	0.00001
Hours/task	3	3	3	3	3	Hours/task	0.50	0.75	1.25
Tasks	250	230	150	75	100	Tasks	250	272	283

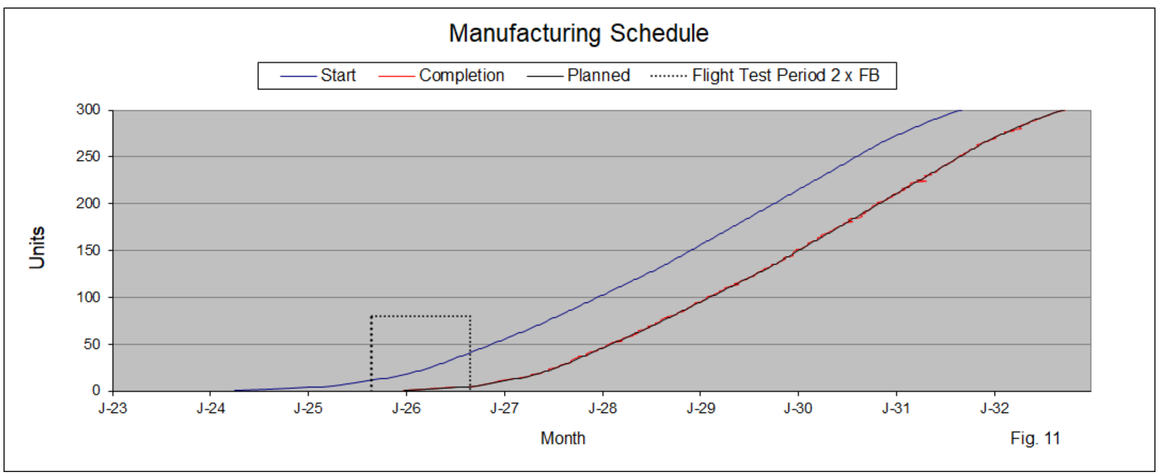


Fig. 11

Figures 12, 13, and 14 show four simulations, black, red, yellow, and blue, each with an increased level of feedback.

The black lines show the engineering and manufacturing headcounts and learning curve when there is no feedback. The engineering effort is completed on 6/1/26 on the planned schedule. The learning curve slope is 71.6%.

The red lines have feedback initiated only in manufacturing. This feedback produces design changes in engineering, generating sustaining engineering. These design changes are fed back into manufacturing resetting its task's current p(t) to their p(0) values. The increased resources are seen in the red line of the manufacturing headcount chart and the flattening of the learning curve.

The yellow lines show the effect of including the internal engineering feedback.

In a project that stretches the engineering capability, due to new technology, inexperience, or other causes, the design process may generate errors. These errors can be expected to show up in the later stages of the first aircraft's production and during the flight test period.

The blue lines illustrate the effect of those errors being doubled between 4 months before the first aircraft's completion and the completion of the engineering test effort. The simulated learning curve has the hump often seen in these circumstances. It is followed by a steep decline and then a transition to a more typical curve.

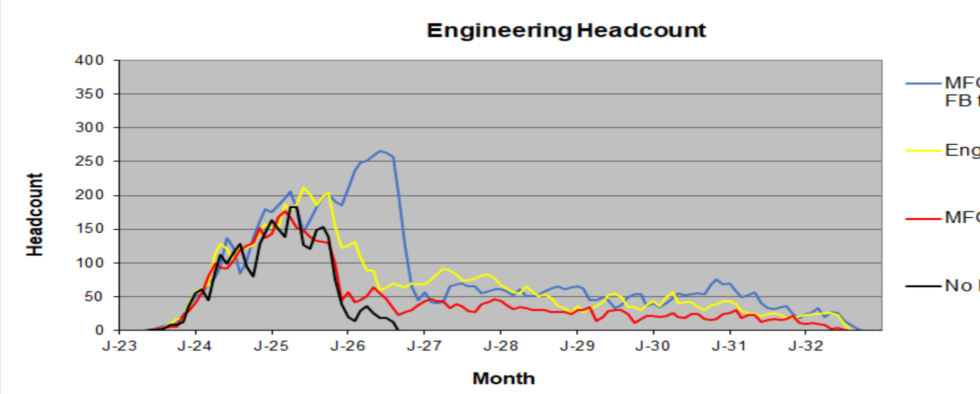


Fig. 12

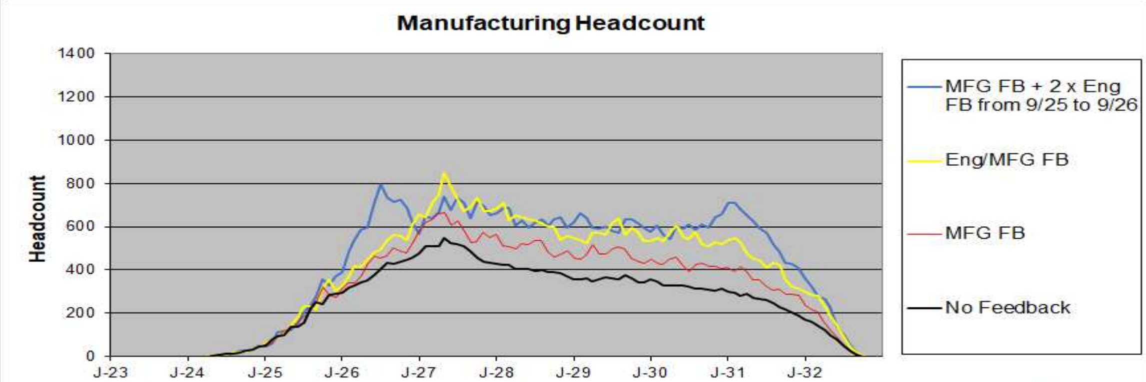
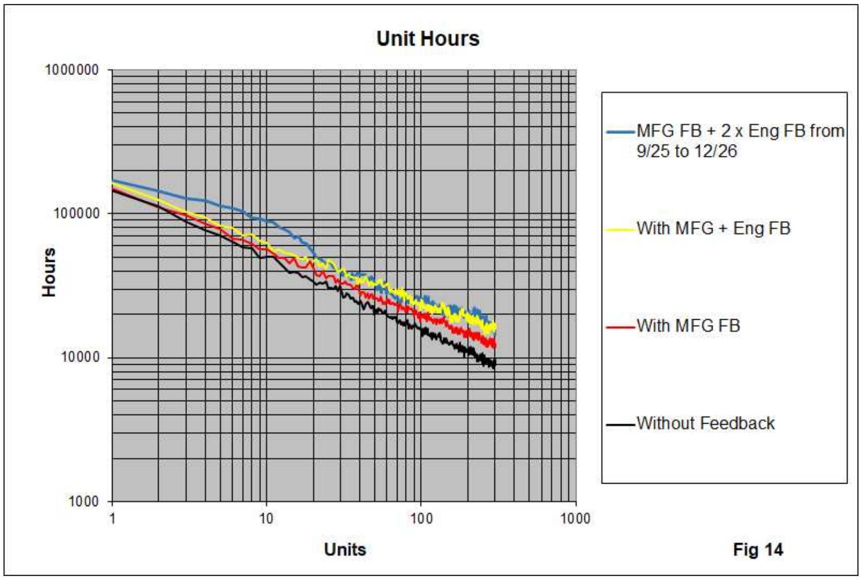


Fig 13



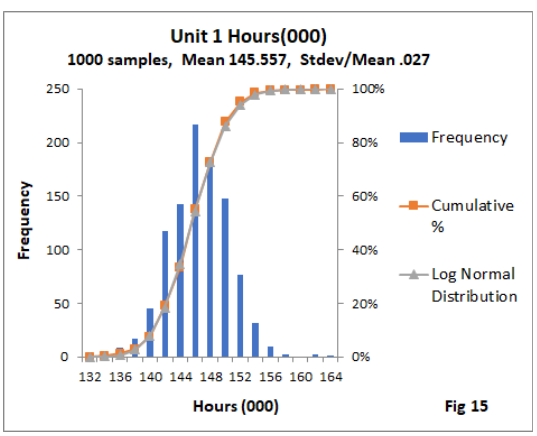
Uncertainty

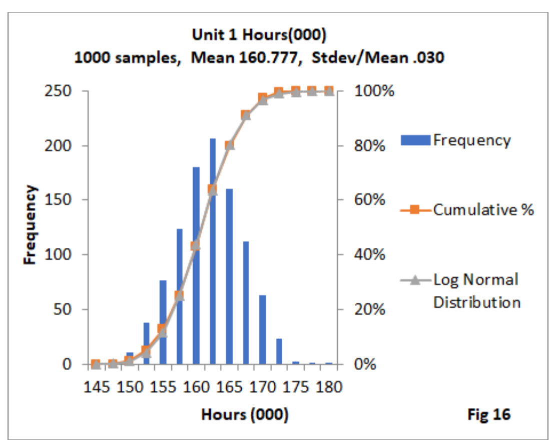
When this stochastic process is implemented as a Monte Carlo system the predictions are in the form of Probability Density Functions, PDFs. In the examples above there are two fundamental causes of variation, the uncertainty of the trial outcomes and the feedback processes.

The histograms below are from 1000 iterations of the time domain model. Figures 15 and 17 show the PDFs of units 1 and 300 hours with only the trial outcome uncertainty, **p(t)** > **x** ~U(0,1). Figures 16 and 18 show the PDFs including the feedback process while holding all of the input parameters constant.

Feedback from design changes increases the variation as the project progresses. Figure 18 shows about twice the variation and hours as the corresponding simulation without design changes, figure 17.

All of the distributions are close to lognormal.





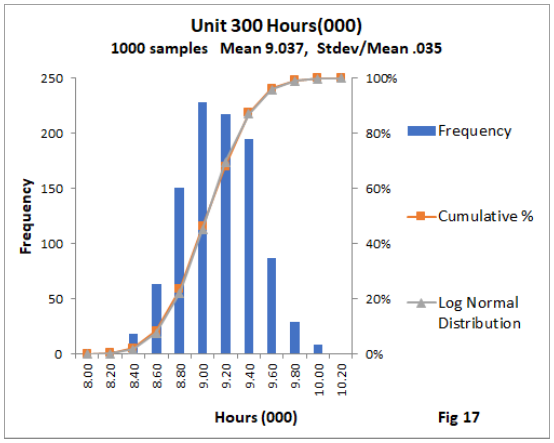


Fig 17

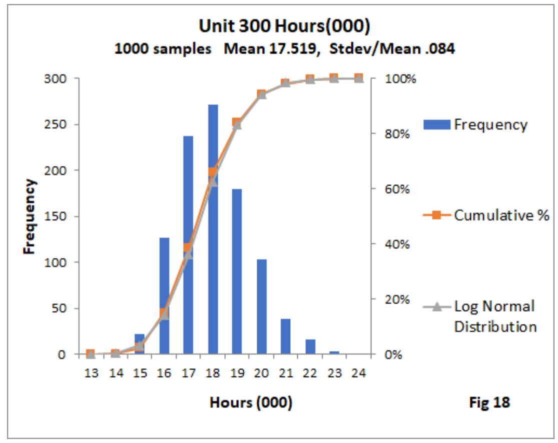


Fig 18

Fixing the Monte Carlo model's random number generator to a single sequence for the first 6 years of the simulation produces learning curves with its uncertainty beginning at unit 96, shown in Fig 19, with expanded scales in Fig 20.

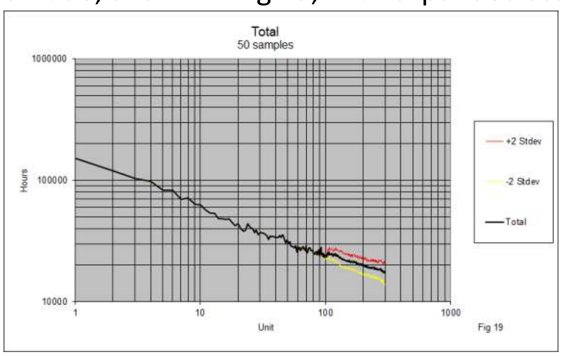


Fig 19

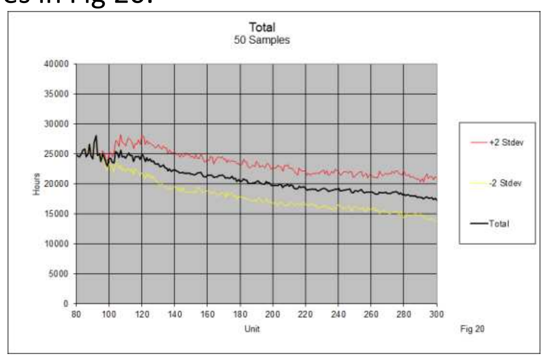


Fig 20

Notably, the uncertainty does not grow as one might expect from a typical random walk model. When the iterations of **p(t**), and thus labor hours**,** to successful completion, are greater than 1/**p(t)**, **p(t)** becomes larger than expected. For the next unit's iterations that larger probability reduces the expected number of iterations. Thus, a slight autoregressive dynamic is produced.

Summary
The iteration of **p(t+1) = p(t) + (1 - p(t)) dl** describes much of the dynamics of the learning curve and work in general. With **N**, the number of tasks, and **p**, a measure of difficulty, both the size and complexity of a project can be modeled. By imbedding the iteration of **p(t)** into a feedback system the impact of the broader work system can also be evaluated. The model is a stochastic process and thus produces a PDF. From the PDF summary statistics can be calculated, but also options that bifurcated the PDF can be evaluated.

The model links work to information. Figure 22 shows the bits per successfully completed unit for the two learning curves in figure 21. Information is calculated as:
I(**unit**) = $(-\sum_{i=t}^{t+n-2} \log_2(1-\mathbf{p}(i))) - \log_2(\mathbf{p}(t+n-1))$,
where t is the first trial of the **unit** and ***n*** is the number of trials to the successful completion of the **unit.** Both curves have 1000 tasks and a trial time of 1 hour. The black

curve, with p(0) = .01 and dl = .00017, has a slope of .717. The blue curve, with a p(0) = .15 and dl = ..003 was chosen to illustrate a curve with a hump that flattens as **p**(large) approaches one. Connecting work with information allows the mathematics of information theory to be brought to bear on the nature of work.

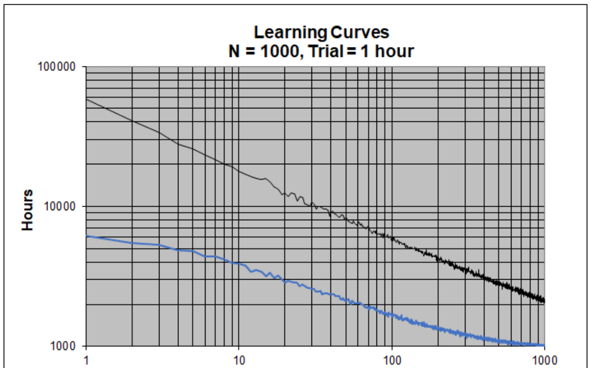


Fig 21

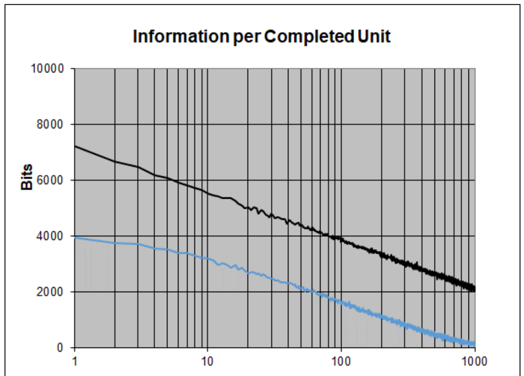


Fig 22