Sequential Testing in Product Development

Stefan Thomke • David E. Bell
Harvard University, Graduate School of Business Administration, Boston, Massachusetts 02163
sthomke@hbs.edu • dbell@hbs.edu

A fundamental problem in managing product development is the optimal timing, frequency, and fidelity of sequential testing activities that are carried out to evaluate novel product concepts and designs. In this paper, we develop a mathematical model that treats testing as an activity that generates information about technical and customer-need related problems. An analysis of the model results in several important findings. First, optimal testing strategies need to balance the tension between several variables, including the increasing cost of redesign, the cost of a test as function of fidelity, and the correlation between sequential tests. Second, a simple form of our model results in an EOQ-like result: The optimal number of tests (called the Economic Testing Frequency or ETF) is the square root of the ratio of avoidable cost and the cost of a test. Third, the relationship between sequential tests can have an impact on optimal testing strategies. If sequential tests are increasing refinements of one another, managers should invest their budgets in a few high-fidelity tests, whereas if the tests identify problems independently of one another it may be more effective if developers carry out a higher number of lower-fidelity tests. Using examples, the implications for managerial practice are discussed and suggestions for further research undertakings are provided.

1. Introduction
A fundamental problem in managing product development is the optimal timing, frequency, and fidelity of sequential testing activities, or tests, that are carried out to evaluate novel product concepts and designs. Testing is an important activity: Studies have shown that it can account for nearly half of total development effort (Shooman 1983). But determining an optimal testing strategy is a difficult task as there is an important tension to be managed. Test models such as prototypes can be expensive, so firms try to lower cost by delaying prototype evaluations as long as possible. It is not unusual for corporate testing departments to receive product prototypes near the end of a development process and conduct a big “killer” test (Reinertsen 1997) to resolve most technical uncertainty. But the same departments also discover that the cost of design changes increases with time: Problems discovered near the end can have serious consequences for the economics of a development project. Thus, a fundamental management problem is how to balance the additional cost of earlier testing with the value of information that early testing can provide.

Suppose that a manager has a budget of $100,000 and needs to decide whether to spend it on frequent testing or on a few tests using higher-fidelity prototypes. Which of the two options is better and when should the tests be carried out? In this paper, we answer the following questions: How often, when, and at what fidelity should testing activities be carried out, given the underlying economics and properties of the tests being planned?

The academic literature provides some help in formulating optimal testing strategies. Wheelwright and Clark (1992) observed that firms often delay prototype testing for too long and thus propose an
approach of “periodic prototyping” where test prototypes are built on a regular schedule, with increasing degrees of fidelity (for example, every other month). Cusumano and Selby (1995) found that software developer Microsoft uses a process they call “synchronize and stabilize” where software prototypes are built and tested on a very frequent basis—ranging from a “daily build” to waiting weeks between tests. While the empirical evidence suggests that early and frequent testing is desirable in some projects, it will certainly not be the most effective strategy for all projects and for all design problems. For example, one would not expect automotive firms to test full-scale prototypes on a daily basis unless, of course, the cost of such prototypes can be reduced dramatically.

In this paper, we treat testing as a sequential activity that generates information about technical problems and/or customer needs. We develop a mathematical model that treats product development as a process of managing the evolution and resolution of uncertainty with the aid of such repeated testing activities within a single test method. An analysis of our model results in several findings, including the following:

- Optimal testing strategies depend on the increasing cost of redesign, the cost of a test as a function of fidelity, and the degree of correlation between sequential tests.
- A simple form of the model results in an “EOQ”- (Economic Order Quantity) like result: The optimal number of tests is the square root of the ratio of avoidable cost and the cost of a single test. We call this number the Economic Testing Frequency or ETF.
- The relationship between sequential tests can have an impact on optimal testing strategies: If sequential tests are refinements of one another, managers should invest their budgets in a few high-fidelity tests, whereas tests that identify problems independently are more effective if developers use their budget on a higher number of lower-fidelity tests. In most cases, however, the ETF formulation developed in this paper provides a good estimate for the optimal number of tests to be carried out.

The plan of the paper is as follows. We start with a general discussion of testing and its costs and benefits (§2). Then (§3) we develop the base model and its assumptions derived from empirical field observations. In §4, we analyze optimal testing through stylized models of increasing degrees of generality and propose different testing strategies. We conclude (§5) by summarizing our findings and propose further research undertakings.

2. The Role of Testing in Managing Uncertainty

Building and testing physical and virtual prototype models is an integral part of product development. This observation is shared by a number of researchers that have studied prototyping in the product design and development process (Bowen et al. 1994, Leonard-Barton 1995, Ulrich and Eppinger 1994, Wheelwright and Clark 1992). Thus Wheelwright and Clark (1992) observed that “senior managers, functional heads, and project leaders who do not fully understand and fully utilize the power of prototyping unintentionally handicap their efforts to achieve rapid, effective, and productive development results” (pp. 255–256).

Models used in testing come in many forms and can be used to reduce uncertainty about customer needs (perhaps by asking customers to evaluate early prototypes in focus groups) and the feasibility of alternative technical solutions (perhaps by testing functional prototypes under laboratory conditions). As a consequence of these uncertainties, all development projects will have problems that need to be detected and solved. In this paper, we examine optimal testing strategies for managing the resolution of such problems in the most economical way.

2.1. The Cost of Repeated Testing

A test’s cost typically involves the cost and time of using equipment, material, facilities, and engineering resources. This cost can be very high, such as when a prototype of a new car is used in destructive crash testing, or it can be as low as a few dollars, such as when a chemical compound is used in pharmaceutical drug development and is made with the aid of combinatorial chemistry methods (Thomke et al. 1998). The cost and time to build a test prototype depends highly on the available technology and the degree of accuracy, or fidelity, that the underlying model is intended to
have (Bohn 1987, Wall et al. 1991). For example, building a physical prototype used in automotive crash tests can cost hundreds of thousands of dollars and may take months to build, whereas a lower-fidelity “virtual” prototype built inside a computer via mathematical modeling can be relatively inexpensive and generated in a matter of hours or days after the initial fixed investment in model building has been made.

Developers often build inaccurate or incomplete prototype models of the final product and/or construct laboratory environments that model conditions of product operation. The value of using such “incomplete” models and laboratory environments is both to reduce investments in aspects of “reality” (i.e., the real product being used under real conditions) that are irrelevant for the test outcome and to control out some aspects of reality that would affect the test to simplify the analysis of test results. Thus, a model of an airplane used in wind tunnel experiments has no internal design details—these are both costly to model and (mostly) irrelevant to the outcome of wind tunnel tests. But sometimes a model to be built is incomplete because one cannot economically incorporate all relevant aspects of the “real” or does not know them. For example, building a high-fidelity prototype early in a development process may not be feasible because the necessary equipment is not available or has not been built yet. The incompleteness of a prototype model and/or test environment leads to residual uncertainty that cannot be resolved until the given model(s) being used in testing is replaced by a different (and more accurate) prototype or by the real product in the real environment for the first time.

Aside from finding design problems, early testing can also be used by marketing departments to reduce uncertainty about customer needs. High-fidelity studies of such customer needs may involve accurate prototypes as described earlier but may also involve large sample sizes of a potential customer population. In contrast, smaller in-depth focus groups or “simulated-use” environments cost less and reduce less uncertainty, but may be a cost-effective way of managing rapidly evolving customer needs. Thus, the fidelity of a test is not only driven by the accuracy of a prototype model shown to customers but how closely the test environment represents the often wide range of real customers in real-use environments.

2.2. The Benefits of Early Information
Tests can generate information about the functionality and manufacturability of a product design and, if evaluated by customers, information about the extent to which it meets customer needs. As the information from testing becomes available to developers (in the form of technical problems or mismatches between product solution and customer needs), it reduces uncertainty and is typically followed by corrective design changes (“redesign”). It is also well known in the product development literature that, everything else being equal, the average cost and time of design changes increase as one nears product launch: Specifications and manuals have to be rewritten, engineering drawings have to be redrawn, expensive tools may have to be scrapped or remachined, etc. Thus, the further a project has advanced, the more resources have typically been committed to a particular solution path and changes at the end are orders of magnitude more costly than changes at the beginning (for empirical findings in software development, see Boehm 1991). So it is not only the information in itself that provides value, but the timing of information generation as it can change the economics of a development project—the earlier new information becomes available, the higher its value (Krishnan et al. 1997, Terwiesch et al. 1998, Thomke and Fujimoto 1999).

The advantages of early information from prototype tests has been demonstrated by a range of studies, including a controlled experiment by Boehm et al. (1984) in which seven development teams developed the same application software following two different processes: (a) specifying where no intermediate prototype was built, tested, and shown to users, and (b) prototyping where a prototype was built and shown to users for feedback about halfway through the ten-week project. The study found that prototyping resulted in products with roughly equal performance but required 40% less software code and 45% less effort. In addition, prototyping resulted in software that rated higher on dimensions that users cared about (e.g., ease of use) but was rated somewhat lower on
functionality and robustness. Among several conclusions, the authors suggest that “prototyping is not necessary on familiar projects where there is little risk of getting the wrong user interface, requirements, or design.” (Boehm et al. 1984, p. 88). This conclusion supports the notion of value of information as a driver for testing strategies: If there is little uncertainty to be resolved, it may be more economical to conduct very few tests, or one big test at the end of a development process. Taken together, one of the most important benefits of earlier testing is a reduction of the total economic cost of design changes which, as described, increase with development schedule.

2.3. The Fundamental Testing Problem
Balancing the cost of repeated testing and the benefits of early information is the managerial problem that we will address in this paper. Considerable benefits from early information call for more frequent and accurate testing activities. In contrast, significant testing costs drives managers towards infrequent and late tests—similar to the big “killer” test described by Reinertsen (1997)—to economize total testing expenses. The solution to this tension is a testing strategy that strikes an optimal balance: the timing, frequency, and fidelity of testing activities that result in the most favorable product development economics.

How are firms currently managing this tension? The literature on prototyping and testing suggests that many firms are testing too infrequently and too late. A possible explanation for this can be found in the way firms account for testing expenses: Modern cost accounting systems can track actual expenses by test activity, whereas the benefits from early information are typically not measured or quantified. Not surprisingly, we would expect to see managerial behavior that emphasizes the well-known costs of testing over its little-known benefits. As a result, this paper intends to make contributions at two levels: First, we define and parameterize a fundamental testing problem faced by managers. We believe that identifying how key parameters have to be weighed against each other will bring clarity to the testing problem, particularly to managers who are currently rethinking their testing strategies. Second, we will make specific recommendations with respect to optimal testing strategies in different model settings of increasing generality and show that such a strategy ought to depend, among several variables, on the correlation between sequential tests—an insight that results directly from the mathematical model.

3. The Base Model
To develop a theory about how testing can remove problems most effectively over the course of a project, we devise a framework for optimal testing strategies. While the framework can be used to model any situation to the degree of complexity required, we will analyze some stylized models of increasing complexity in §4 to understand the general implications of the framework.

3.1. Assumptions
At the core of our approach is the idea that uncertainty about a given product attribute can arise as a result of poor understanding of customer expectations about quality, features, and price (i.e., customer uncertainty) or because of the difficulty in predicting the feasibility of a technical solution prior to testing (i.e., technical uncertainty). As a consequence of customer and technical uncertainty, all development projects will have problems. The objective of testing is to identify design problems so that they can be solved through rework. As a project progresses, undetected problems will accumulate until a test reveals them (or a fraction of them if the test is of less than full fidelity). More specifically, we assume the following:
• A project gets underway at time $t = 0$ and the number of cumulated problems at time $t$ is $v(t)$. These problems could be due to technical errors (Does it work?), mismatches with customer needs (Does the customer like it?), and manufacturing issues (Can it be produced efficiently?).
• The quantity $v(t)$ is really a measure of the amount of rework that would be required at time $t$ if all detectable problems were discovered and no earlier rework had been completed. Of course $v(t)$ will never be completely known until after the project is over. For example, thorough testing at time $t$ may not identify all problems; these might only be caught after the product is launched or when customer complaints are reported.
For simplicity, we assume that at the outset of a project, \( v(t) \) may be estimated. Remember \( v(t) \) measures when problems become detectable—not when they are actually identified (which depends on the testing schedule). Thus, design problems might well be detectable at \( t = 0 \), so that \( v(0) > 0 \), but these might not actually be identified until the end of the project (which we will assume is \( t = 1 \)).

- Problems are homogenous with respect to the cost of solving them at a given time \( t \) (i.e., each discovered problem faces the same per unit rework cost). We further assume that solving problems is more costly towards the end of project than it is at the beginning, so a design problem that is not noticed for quite some time will be more costly to fix than if it had been identified at the outset. We will assume that the “per unit” cost of rework is \( d(t) \) at time \( t \). So, for example, if no testing is done until time \( t \), at which point all \( v(t) \) problems are discovered, it will cost a total of \( d(t) v(t) \) to fix them. The remaining \( v(1) - v(t) \) problems, if not detected until the end of the project, will cost a further \( d(1)(v(1) - v(t)) \) to repair.

- We further assume that \( v(t) \) increases over the course of the project and that testing activities themselves do not introduce new problems. It is likely that problems will be discovered and solved in the normal course of project without any formal testing. It might be thought that therefore \( v(t) \) could actually decrease. Even if problems are discovered fortuitously, the rework expense must still be incurred and we believe it will be more accurate to ignore this effect.

- Tests can be carried out with the aid of “incomplete” prototypes and/or in laboratory environments where only a fraction of the problems are identified—an attribute that is captured in the variable fidelity \( f \). Thus, a high-fidelity test will tend to uncover most currently detectable problems; a low-fidelity test will catch only a few. Low-fidelity tests will have the advantage of being cheaper and quicker—a fiberglass model rather than metal, a focus group instead of a mass survey. It may also be that full-fidelity tests (tests that uncover all current problems) are impossible at an early stage of the project, or at least are prohibitively expensive, which is a constraint that we will revisit in §4.4.

- We also assume that at the end of the project, \( t = 1 \), a full-fidelity test is carried out. One may think of this as simply the test of the market place—problems are either corrected at launch or are paid for indirectly through lost sales. All this is assumed to be incorporated in the quantity \( d(1) \).

- The cost of testing can be incurred in two ways: out-of-pocket cost (expenses), and through time delays caused by performing the test and the resultant rework. Delays are inevitable once problems are created, unless overtime is possible, which translates the time problem into an expense. For simplicity we ignore the effect of delays and treat the cost of a test as being purely financial. We do examine the case where available testing strategies change over time (e.g., full-fidelity tests are not feasible at the beginning of a project).

- Generally, we will assume that full-fidelity tests cost the same amount \( m \) per test, whenever they are conducted. Experience suggests that tests will be less costly at earlier stages in the project, but this may be because earlier tests tend to have lower fidelity; the tests are cheap because they are low fidelity, not simply because they are early. Constant \( m \) may be a better approximation for marketing tests (focus groups cost the same no matter when conducted) than for physical prototypes, as a high-fidelity test can be very expensive (or impossible to construct) at an early development stage. Our assumption that test costs are independent of time is thus an approximation. Furthermore, the assumption (\( m \) being constant) is realistic during phases in which the basic testing method remains the same. Across phases, however, the model may not hold.

- To summarize, though the cost of a test might depend upon both time and fidelity, we will assume in this model that it depends only on fidelity, \( m(f) \). We will usually assume the simple form \( m(f) = mf^\gamma \) as this captures the reality that the last 10% of fidelity is harder to achieve than the first 10%. At times we generalize the formula to \( m(f) = mf^\gamma \); the higher \( \gamma \) is, the more accelerated the costs as a function of fidelity. The main variables of our model are summarized in Table 1.

Table 1.

It is our intent to use the model elements presented to analyze stylized, but nonetheless fundamen-
tal, issues in testing and develop general insights that can be used to formulate qualitative suggestions for managers facing development decisions.

3.2. Costless Testing

If testing were costless, then so long as \(d\) and \(v\) are increasing, it would make sense to test and repair continuously at a total cost of \(\int_0^1 d(t)v(t)dt\). This is the ideal case in which problems are detected and solved as soon as they occur. If \(d(t) = dt\) and \(v(t) = vt\) then the “no test” cost is \(dt\), whereas the best possible “continuous redesign” option would cost \(\int_0^1 dvdt = dv/2\). There is a limit to how much of the rework cost can be eliminated by even the most effective testing strategy. We will refer to the difference between the no test cost and the continuous testing cost as the project’s avoidable cost. Note that if \(d(0) > 0\), then the quantity \(d(0)v(1)\) is unavoidable. Thus any optimal testing strategy must be independent of \(d(0)\), which we may therefore assume to be zero, without loss.

3.3. Relationship Between Sequential Tests

A final but important element of the framework concerns the relationship between two or more different tests. A full-fidelity test at time \(k_1\) resolves all \(v(k_1)\) of the problems. A subsequent full-fidelity test at time \(k_2\) resolves all problems that occur between times \(k_1\) and \(k_2\), in amount \(v(k_2) - v(k_1)\). But what should we assume about the relationship between two or more low-fidelity tests? Do they tend to cover the same pool of problems? In general, if the first test of fidelity \(f_1\) resolves \(f_1v(k_1)\) of the problems, then the second test of fidelity \(f_2\) will resolve \(f_2v(k_2) - \alpha f_1v(k_1)\), where \(\alpha\) depends on the relationship between two tests.

For the purpose of analysis, the following three cases represent a spectrum of possible relationships between tests (for an illustration, see Figure 1). A second test may be:

- **Fully overlapping** (\(\alpha = 1\)) with the first test and resolve only \(f_2v(k_2) - f_1v(k_1)\) of the currently undetected problems. That is, the problems identified in the second test include all of those found in the first test. This may occur if the second test is simply an update of the first, as, for example, in the switching to a more realistic prototype and/or laboratory conditions—a scenario that is very common in product development.

- **Partially overlapping** (\(0 < \alpha < 1\)) with the first test. A special case is statistical independence when \(\alpha = f_2\) where the second test would resolve \(f_2v(k_2) - f_1v(k_1)\) of the current problems. Only a fraction \(f_2\) of the problems identified earlier could be “rediscovered.” This may occur if a different kind of test is conducted, or market research is performed with a different set of customers.

- **Complementary** (\(\alpha = 0\)) to the first test and resolve \(f_2v(k_2)\) of the problems, or \(v(k_2) - f_1v(k_1)\), whichever is smaller. This might occur if the two tests address two different problem categories. Such a case can be regarded as separable (into different problem categories) for our purposes and thus can be treated as a much simpler form of the above cases.

---

**Table 1** Summary of Base Model Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(t)</td>
<td>development project timeline (0: start; 1: end of project)</td>
</tr>
<tr>
<td>(v(t))</td>
<td>number of cumulated problems at time (t)</td>
</tr>
<tr>
<td>(d(t))</td>
<td>“per unit” cost of rework at time (t)</td>
</tr>
<tr>
<td>(f)</td>
<td>fidelity of test ((0 &lt; f &lt; 1)) measures the fraction of cumulated problems detected</td>
</tr>
<tr>
<td>(k)</td>
<td>timing of a particular test ((0 &lt; k &lt; 1))</td>
</tr>
<tr>
<td>(m(f))</td>
<td>cost of a test as a function of fidelity</td>
</tr>
<tr>
<td>(ETF)</td>
<td>optimal number of tests (or Economic Testing Frequency)</td>
</tr>
</tbody>
</table>

**Figure 1** Different Relationships between Tests (Static Case: \(\nu\) Remains Constant between First and Second Test)

- **Fully Overlapping**
  - \(f_1 = 0.25\)
  - \(f_2 = 0.50\)
  - \(t_2 = t_1 + \frac{g}{2}\) with the first test.

- **Partially Overlapping**
  - \(f_1 = 0.25\)
  - \(f_2 = 0.50\)
  - \(t_2 = t_1 + \frac{g}{2}\) with the first test.

- **Complementary**
  - \(f_1 = 0.25\)
  - \(f_2 = 0.50\)
  - \(t_2 = t_1\) with the first test.
To illustrate the relationships between tests, consider the following two examples:

**Example 1.** Resolving technical uncertainty in the development of integrated circuits.

Modern day digital integrated circuits such as microprocessors can contain millions of individual circuit components (known as “gates”) and firms can employ hundreds of developers that are involved in the design of such circuits. Repeated testing using simulated or physical models of a circuit design plays an integral role in the development of integrated circuits. To simplify our example, suppose our circuit contains a set of \( m \) inputs, \( n \) outputs, and \( x \) controls. (It is not unusual for \( m + n + x \) to exceed one hundred pins in circuits of moderate complexity.) To test a circuit, one would try to verify whether an input pattern of binary signals maps onto a desired output pattern, given a configuration of control inputs. As the number of possible permutations is too large for exhaustive testing protocols, optimal testing strategies matter a great deal in the identification of design problems. During each test, a family of random input patterns is applied to a test model of the circuit. Suppose the number of possible permutations is one billion—a small number compared to real-world testing—and the family of input patterns represents one million of these permutations, then the test’s fidelity with respect to input patterns is 0.001, or 0.1%. As repeated tests apply input patterns selected at random to a prototype circuit, it would be an example of *partially overlapping* testing.

Similarly, circuit models are likely to be improved over the course of a project and, as a result, the tests described above can be carried out with more accuracy. For example, simple wire-wrapped breadboards are replaced with printed circuit boards, etc. It a new circuit prototype is simply an improvement of an earlier version and identifies the same category of problems but with higher-fidelity, repeated tests with such an improved prototype—but the same input pattern—would be considered *fully overlapping*. Sometimes circuit prototypes are designed to identify problems from different categories and thus do not overlap. Repeated tests with such prototypes would be considered *complementary* as they complement each other during development. For example, consider that circuit designers often build (virtual or physical) test models that aid in the identification of logic, timing, and signal quality problems (Thomke 1998). Simulation models that are very good at identifying logic problems may not be able to identify any timing or signal quality problems and thus do not overlap with models that are specifically designed to identify such timing and signal quality problems. In this paper, we would regard the overall testing strategy as separable and would develop three complementary testing strategies for each of these three problem categories.

**Example 2.** Resolving uncertainty about customer needs in the development of software.

Research has shown that familiarity with existing product attributes can interfere with an individual’s ability to express needs for novel products (von Hippel 1988). In other words, lack of experience in the use of a product (which is normally the case with new products) diminishes the user’s ability to accurately describe needs. Needs become more refined (or change) as the user comes in direct contact with the product and engages in problem solving. This happens quite often in systems that involve human-machine interactions, resulting in responses such as “I’m really not sure what I want, but I’ll know when I see it.” For example, designers of applications software sometimes find that customers significantly revise requirements after they use software for the first time, leading to very costly and time-consuming redesigns of an otherwise functional product.

To resolve uncertainty about customer needs, software developers have started to show early prototypes to groups of customers and ask for feedback on specific product attributes. As we have seen in §2.2., such early feedback has proven to be very valuable in the development of novel applications software. Suppose that during each test, a group of \( n \) randomly selected customers evaluates a software prototype of fidelity \( f \) with respect to a category of potential problems. These \( n \) customers represent a larger group of customers in a given market segment and thus constitute a sample—similar to the family of input patterns selected at random in the circuit example described earlier. Thus, during repeated testing each customer sample would be *partially overlapping* with respect to earlier samples.

As development progresses, a given customer sample can be exposed to an improved software prototype. Similar to the first example, repeated testing with
such prototypes can be regarded as fully overlapping if increasingly higher-fidelity product prototypes are shown repeatedly to the same set of customers.

4. Model Analysis

In this section, we will analyze a set of stylized models with increasing degrees of generality (for an overview of models analyzed in this section, see Figure 2). The purpose of the analyses is to develop some robust insights into the nature of optimal testing as opposed to exact numerical results for specific testing problems. To strengthen our analysis, we have derived closed-form solutions wherever possible and examined them under a range of assumptions.

We begin with two simple but practical cases of a single full-fidelity test and a single variable-fidelity test. Analyzing these cases helps us to build the foundations for the more general multiple test cases that are analyzed next. Next we solve the multiple full-fidelity test case which results in a functional form that is reminiscent of the well-known Economic Order Quantity (EOQ) formulation. We conclude the section by analyzing the most general case, multiple variable-fidelity tests, under the conditions of partially and fully overlapping between tests. In all cases, we seek the testing strategy that minimizes total cost: the sum of rework and the cost of testing.

4.1. A Single Full-Fidelity Intermediate Test

Let us suppose we intend to perform only one intermediate full-fidelity test (recall there is a “compulsory” test at \( t = 1 \)). It would be good to do the test early, so as to perform rework when it is less costly. On the other hand, the longer we wait, the more problems we can identify and correct. In the simplest case, when \( d(t) = dt \) and \( v(t) = vt \), a test carried out at time \( k \) will incur a redesign cost of \( d(k)v(k) = dvk^2 \) at time \( k \) and a redesign cost of \( d(1)v(1 - v(k)) = dv(1 - k) \) at time 1. The total cost is \( dv(k^2 + 1 - k) \), which is minimized when \( k = 1/2 \). A slightly more general case, when \( d(t) = d_0 + d_1t \) and \( v(t) = v_0 + v_1t \) is optimized when \( k = (1 - v_0/v_1)/2 \) and \( v(k) = (v_0 + v_1)/2 = v(1)/2 \) (if \( v_0 > v_1 \), then the test should be performed at time 0). Note that this solution is independent not only of \( d_0 \) but also of \( d_1 \). Indeed, whenever \( d(t) \) and \( v(t) \)
are linearly related, say \( d(t) = a + bv(t) \) for some constants \( a \) and \( b \), then the optimal timing of a single test is when \( v(k) = v(1)/2 \) (and, for \( n \) such tests, when \( v(k) = iv(1)/(n+1) \) for \( i = 1, \ldots, n \). The idea that \( d(t) \) might be closely related to \( v(t) \) is not so unlikely. The more problems that need to be corrected, the more interlinkages there will be.

Even when \( d(t) \neq a + bv(t) \) for any constants \( a, b \), the result that \( v(k) = v(1)/2 \) is surprisingly robust. For example, if \( d(t) = dt^a \) and \( v(t) = vt^b \) (for various parameters \( d, v, a, \) and \( \beta \)) then the best time to perform a test is when \( k = [\beta/(\alpha + \beta)]^{1/\alpha} \), a value that varies from 0 to 1 depending on \( \alpha \) and \( \beta \). However, \( v(k) = v(1)[\beta/(\alpha + \beta)]^{\beta/\alpha} \), while not always \( v(1)/2 \), is relatively constant for a broad range of values of \( \alpha \) and \( \beta \).

The following two assumptions are therefore closely related from the point of view of our analysis.

Assumption 1. \( d(t) = dt, \ v(t) = vt \).

Assumption 2. \( d(t) = a + bv(t) \) for some constants, \( a, b \).

To see the equivalence, note that in the second case, by a change of scale from “\( t \)’’ to “\( a + bv(t) \)”, we reproduce the first assumption. Because the demands on notation will be much simpler if we take \( d(t) \) and \( v(t) \) to be linear (\( d(t) = dt \) and \( v(t) = vt \)), we will do so in the remainder of the paper. Our results are more general, however, due to the reinterpretation to the second case above. For example, under Assumption 1, three intermediate tests would be conducted at \( t = \frac{1}{4}, \frac{1}{2}, \frac{3}{4} \). Under Assumption 2 they would be carried out when \( v(t) \) equals \( v(1)/4, v(1)/2 \) and \( 3v(1)/4 \).

We will follow a practice of summarizing our findings in the form of “results.” These are formulation specific but indicative of broader rules of thumb.

Result 1. For a single full-fidelity intermediate test, the optimal timing is about halfway through the project or after about half of the total problems have arisen.

4.2. A Single Variable-Fidelity Intermediate Test

If a single low-fidelity test is carried out at time \( k \) then the total cost will be

\[
mf^2 + d(kv) + d(v - kv),
\]

the three terms being the cost of the test, the cost of intermediate redesign, and the cost of final redesign. Differentiating with respect to \( k \) and \( f \), respectively, we get \( 2mf - df = 0 \) and \( 2mf - d(k(1-k)) = 0 \) with solutions \( k = \frac{1}{2} \) and \( f = dv/8m \). (Second-order conditions show this is, indeed, a local minimum; though we will not report it, this is true for of all solutions that we give from here on.) Again we see that the test is carried out at \( k = \frac{1}{2} \). A low-fidelity test is called for if \( dv/8m < 1 \) or \( dv/2m < 4 \). This latter observation will make sense in §4.3 when we see that the optimal number of full-fidelity tests is \( \sqrt{dv/2m} \) including the final test at \( t = 1 \). Thus \( dv/2m < 4 \) corresponds to less than 2 full-fidelity tests, indicating that, other than the final test, less than a full intermediate test is called for.

Result 2. For a single variable-fidelity test, the optimal timing remains about halfway through the project. The option of a low-fidelity test is relevant only if \( m > dv/8 \), that is, if the cost of a test is more than 25% of the avoidable cost.

4.3. Multiple Full-Fidelity Tests

If \( n \) full-fidelity tests are to be carried out (at \( t = 1/n, 2/n, \ldots, n/n \) which is a simplifying assumption in general) the redesign work for each test will be exactly \( v/n \) (being the number of new problems introduced since the last test), and these will cost \( kd/n \) to repair at time \( k/n \). The total cost is thus \( \sum_{k=1}^{n}(m + (kd/n)(v/n)) = mn + dv(2 + dv/2n) \) (using the fact that \( \sum_{k=1}^{n}k = n(n + 1)/2 \)). This is minimized when \( n = \sqrt{dv/2m} \), with a cost of \( dv/2 + 2mdv \). The fraction of the avoidable cost that remains is \( \sqrt{8m/dv} \).

The notion that the optimal number of tests is related—in a simple way—to the ratio of avoidable cost \( dv/2 \) and the cost of testing \( m \) is appealing. The square root form is reminiscent of the familiar Economic Order Quantity (EOQ) inventory model. The analogy between the two physical models can be seen if one thinks of the “problems” in our model as “demand” and \( m \) as the fixed cost of a reorder. The escalation in \( d(t) \) is the holding cost of inventory: By delaying repair one incurs a holding cost (in that \( d(t) \) increases); too frequent a repair means more fixed payments of \( m \). EOQ-like formulations have been found in other models of product development because of the nature of the tension being
modeled. For example, Ha and Porteus (1995) develop a dynamic program that optimizes the number of design reviews in the concurrent development of product and process design. If process design proceeds without information from product design, firms face downstream rework. Design reviews prevent downstream problems through information exchange between product and process design teams but come at a penalty in the form of a set-up and execution time. The authors find that the optimal review period can be approximated by an EOQ-like formulation, resulting from the tension between rework prevention and design review cost. Similarly, Loch and Terwiesch (1998) develop a mathematical model that investigates the extent to which sequential tasks should be overlapped to minimize time-to-market. The time savings from overlapping must be weighed against the delay from rework that result from proceeding based on preliminary information. Communication can mitigate downstream problems through information exchange between product and process design. If process design proceeds based on preliminary information but comes at an expense. Again, Loch and Terwiesch (1998) find that communication can be approximated by an EOQ-like formulation, and design review cost. Similarly, Loch and Terwiesch develop a mathematical model that investigates the extent to which sequential tasks should be overlapped to minimize time-to-market. The authors find that the optimal review period can be considered an activity with “setup” and “holding costs.”

In our formulation, the mathematical connection stems from the fact that we are minimizing a formula that is the sum of linear and reciprocal terms in n. We will see that the simple ratio dv/2m and its square root will recur often in our calculations. We will reserve the symbol n* to mean exactly this square root and will refer to it as the Economic Test Frequency, or ETF, in the remainder of the paper.

**Result 3.** For multiple full-fidelity tests, the optimal frequency of tests is \( n^* = \sqrt{dv/2m} \), the Economic Test Frequency or ETF, which is the square root of the ratio of avoidable redesign cost and the cost of testing.

### 4.4. Multiple Variable-Fidelity Tests: The Fully Overlapping Case

If we conduct n tests with fidelity \( f_1, f_2, \ldots, f_n \) (with \( f_n = 1 \) at \( t = 1 \)), then the amount of rework needed at \( t = i/n \) is \((iv/n)f_i - ((i-1)v/n)f_{i-1}\), assuming this is positive. (If it is not positive we may as well set \( f_i = 0 \), leaving the rework to be done at \( t = (i+1)/n \) as \((i+1)v/n)f_{i+1} - ((i-1)v/n)f_{i-1}\). Assuming \( f_1 \leq f_2 \leq \cdots \leq f_n = 1 \), the total cost is

\[
\sum_{k=1}^{n} \left( mf_k^2 + \frac{kd}{n} \left( \frac{kv}{n} f_k - \frac{(k-1)v}{n} f_{k-1} \right) \right)
\]

\[
= \sum_{k=1}^{n-1} \left( mf_k^2 - \frac{dvk}{n^2} f_k \right) + dv + m.
\]

Note that this sum is separable in the \( f_k \)'s. Differentiating with respect to \( f_k \) gives

\[
2mf_k = \frac{dvk}{n^2} \quad \text{or} \quad f_k = \frac{dv}{2m} \cdot \frac{k}{n^2} = k \left( \frac{n}{n^*} \right)^2
\]

where \( n^* = \sqrt{\frac{dv}{2m}} \).

We may consider three cases:

- If \( n \leq n^* \) then all \( f_k = 1 \), with a total cost of \( mn + \frac{dv}{2} + \frac{dv}{2n} \). As we have seen, the lowest value occurs when \( n = n^* \), and is \( dv/2 + \sqrt{2mdv} \).

- If \( n > (n^*)^2 \) then all \( f_k \) are fractional (since \( f_n = n(n^*/n)^2 < 1 \)). In this case the total cost may be calculated as \( m + dv - (d^2v^2/4nn^*)((n-1)(2n-1)/6) \) (the derivation uses the fact that \( \sum_{k=1}^{n} k^2 = n(n+1)(2n+1)/6 \)). This is clearly increasing in \( n \), so the minimum expense occurs when \( n = (n^*)^2 \), when the cost is \( 5dv/6 + 3m/2 - n^2v/3dv \).

- If \( n^* < n \leq (n^*)^2 \) then the early \( f_k \)'s are fractional, the latter ones are equal to 1. The transition point comes when \( f_k = k(n^*/n)^2 = 1 \). The total cost in this case may be calculated as

\[
mn + \frac{dv}{2} + \frac{dv}{2n} + \frac{m}{2} - \frac{2m^2n^2}{3dv} - \frac{1}{12} \frac{dv}{n^2}
\]

or

\[
m \left( n + n^2 + \frac{n^2}{n} + \frac{1}{2} - \frac{1}{3} \left( \frac{n}{n^*} \right)^2 - \frac{1}{6} \left( \frac{n^*}{n} \right)^2 \right).
\]

Figure 3 shows this piecewise cost function (divided by \( m \)) for an illustrative case, \( n^* = 6 \).

**Result 4A.** For multiple variable-fidelity tests that fully overlap, the optimal number of tests is ETF = \( \sqrt{dv/2m} \), or the integer above or below if \( \sqrt{dv/2m} \) is a fraction. If the ETF is below \( \sqrt{dv/2m} \), then all tests have full fidelity, otherwise all but the first test have full fidelity.
The bottom line of this result is not so much the details of the solution but rather the nature of the solution, namely that, despite the possible attractiveness of low-fidelity tests, the solution is to use primarily full-fidelity tests. (See Appendix for a detailed explanation.)

The Case of Constrained Fidelity. If the fidelity of intermediate tests is constrained (e.g., high-fidelity tests cannot be carried out very early in the project), say \( f_k \leq k/n \), then if the tests continue to cost exactly \( m \) each, the total cost is \( mn + dv/2n - dv/6n^2 + 2dv/3 \). If \( n \) is at all large, the term involving the reciprocal of \( n^2 \) is small and the optimal solution is approximately \( n^* = \sqrt{dv/2m} \). If the cost of a constrained test is \( mf^2 \), then the total cost is \( mn/3 + m/2 + m/6n + dv/2 + 2dv/3 - dv/6n^2 \). Again, ignoring the reciprocal in \( n^2 \), the optimal solution is now \( \sqrt{\frac{1}{3} + 3n^*} \), or approximately \( 1.75n^* \). Note, though, that the “total fidelity” of all tests carried out, \( \sum_{k=1}^{n} f_k = (n+1)/2 \approx (1.75n^* + 1)/2 \) is still about \( n^* \).

The Case of Accelerated Cost of Fidelity. As one might suppose, if the savings to be had from low-fidelity tests is increased by replacing the function \( mf^2 \) by \( mf^\gamma \) for values of \( \gamma \) higher than 2, then there is a greater tendency to use low-fidelity tests. As we show below, however, the lure of full-fidelity tests is still strong. We will consider a class of tests in which all are of constant fidelity \( f < 1 \), except the last, which is, as usual, of full fidelity. The cost of a test is \( mf^\gamma \).

We have seen that the redesign savings for \( n \) full-fidelity tests are \( dv - (dv/2 + dv/2n) = (dv/2)(1 - (1/n)) \). The total savings from our new scenario is therefore \( (dv/2)(1 - (1/n))f - mn(n-1)f^\gamma - m \). Differentiating with respect to \( n \) and \( f \) yields, respectively, \( df/2n^2 = mf^\gamma \) and \( df/2(1 - (1/n)) = \gamma(n-1)mf^\gamma - 1 \). Comparing the implications for \( f^\gamma - 1 \) we find \( f^\gamma - 1 = dv/2m \cdot 1/n^2 \) and \( f^\gamma - 1 = dv/2m \cdot 1/ny \).

If \( n \leq n^* = \sqrt{dv/2m} \), then \( f = 1 \) and the solution is, as usual, \( n = n^* \). If the solution is that \( f < 1 \), then we must have both \( n > n^* \) and \( n = \gamma \). Most importantly, low-fidelity tests are optimal only if \( \gamma > n^* \). To be clear, suppose the optimal number of full-fidelity tests (using a criterion of either \( m \) or \( mf^2 \) for the cost of a test) is \( n^* = 6 \), then even if the cost of a test were changed to \( mf^\gamma \), the full-fidelity solution would be as good or better than any set of constant lower-fidelity solutions.

RESULT 4b. For multiple variable-fidelity tests that fully overlap, the optimal solution is to perform \( \sqrt{dv/2m} \) full-fidelity tests, as long as this number exceeds \( \gamma \), the exponent of the cost function \( mf^\gamma \).

4.5. Multiple Variable-Fidelity Tests: The Partially (Independent) Overlapping Case

Again, let \( f_1, \ldots, f_n \) be the fidelities of tests carried out at \( t = 1/n, \ldots, n/n \), with \( f_1 = 1 \), and the cost of a test, \( mf^2 \). In Period 1, an amount \( (v/n)f_1 \) is resolved, leaving \( (v/n)(1 - f_1) \) unresolved. In Period 2 an amount \( (v/n)(1 + (1 - f_1))f_2 \) is resolved, and so on. In general, if \( \theta_t \) represents the fraction of \( v/n \) that is unresolved prior to the test in period \( k \), then

\[
\theta_1 = 1 \\
\theta_2 = 1 + (1 - f_1) = 1 + (1 - f_1)\theta_1 \\
\theta_3 = 1 + (1 - f_2) + (1 - f_1)(1 - f_1) = 1 + (1 - f_2)\theta_2 \\
\theta_t = 1 + (1 - f_t)\theta_{t-1} \\
\theta_t = 1 + (1 - f_t)\theta_{t-1} \\
\theta_t = 1 + (1 - f_t)\theta_{t-1}
\]

and in general \( \theta_{t+1} = 1 + (1 - f_t)\theta_t \). The total cost (testing and rework) is \( \sum_{k=1}^{n} (mf_k^2 + (dv/n^2)f_k\theta_k) \).

Table 2 shows the optimal solutions to this problem for \( n = 5 \) to 12 assuming \( \sqrt{dv/2m} = 5 \). Of course when \( n = 5 \) all the \( f_5 \)s equal 1. As \( n \) gets larger the \( f_5 \)s drop, but the savings increase. Indeed, the savings increase for all \( n \). Two notable properties of the solutions can be observed from the table and these hold in general

(i) \( f_{k+1} = f_{k+1}^{(n)} \) for all \( k \) and \( n \);
(ii) when \( n \) gets large, \( f_k \) is approximately constant.
This last property suggests that we again consider the case in which all $f_k = f$, except that $f_{n} = 1$. The total redesign savings in this case are $\alpha = \frac{\alpha}{\beta} \left[ f - (1 - f) \sqrt{n} \right] + \frac{2(1 - f)(n^2 f^2)}{\beta}$ less the testing costs of $m(n-1) f^2 + m$. For the sake of comparison Table 3 shows the optimal $f$ values, and net savings for the same scenarios as in Table 2.

Once again, the details are less important than the style of the solution: It pays to do many low-fidelity tests as long as there is no fixed cost of testing.

### 4.6. Sensitivity of Results to Model Assumptions

As in all models, results are in part a function of assumptions. We have tried to make our assumptions as realistic as possible and, as described earlier, grounded in empirical observations. The result from §4.5, however, requires additional analysis because of its significant consequences for testing strategies. The conclusion that for independent tests one should perform as many tests as possible depends on the fact that our formula for the cost of a test, $m f^2$, approaches 0 as $f$ gets small. If we include a fixed cost component, $m(f) = m_0 + (m - m_0) f^2$, we can expect the optimal number of tests to be finite. Indeed, as $m_0$, the fixed cost, approaches $m$, the cost of a full-fidelity test, the attraction of low-fidelity tests gradually disappears. Figure 4 illustrates the percent of avoidable cost removed both as a function of the number of tests performed (horizontal axis) and as a function of $m_0/m$, the percentage of total testing cost that is fixed (shown by the different curves). As in Table 2, the situation analyzed is $\sqrt{m t/2m} = n^* = 5$. The uppermost curve in the graph illustrates the value of doing many tests when there is no fixed component of the test cost. The lowest curve shows the value of doing various numbers of tests when $m_0 = m$, that is, all tests may as well be of full fidelity. In this case it makes no difference whether the tests are fully or partially overlapping, so the curve represents either. The fully overlapping case with 0% fixed cost and the partially overlapping case with 50% fixed cost have very similar payoffs. Finally, the partially overlapping case with 10% of the test cost fixed is illustrated because it is approximately flat after $n = 5$; that is, only if the fixed cost is less than 10% of the total (with these parameters) does it pay to do more tests.
The figure is very instructive because it demonstrates an important conclusion: The solution $ETF = n^* = \sqrt{dv/2m}$ is a robust solution to the question of how many tests to order. The only situation in which more tests are in fact economical is when the fixed cost component of a test is low and the tests partially overlap.

Result 5. For multiple variable-fidelity tests that partially overlap, the solution is to do very frequent testing. This result reflects the absence of a fixed cost component in the cost of testing. As the fixed cost component increases, the optimal number of tests again approaches the Economic Testing Frequency $\sqrt{dv/2m}$.

5. Discussion

In this paper, we have argued that testing is an important driver of product development performance. Using a mathematical model which treats testing as a sequential activity that reduces uncertainty about technical solutions and customer needs, we have found a number of important insights (summarized in Table 4).

- The notion that the optimal number of tests is related—in a simple way—to the ratio of avoidable cost $(dv/2)$ and the cost of testing $(m)$. The square root form $\sqrt{dv/2m}$ is named the Economic Testing Frequency or ETF and is reminiscent of the familiar Economic Order Quantity inventory model. The analogy can be seen if one thinks of the “problems” in our model as “demand” and $m$ as the fixed cost of a reorder. The escalation in $d(t)$ is the holding cost of inventory: By delaying repair one incurs a holding cost (in that $d(t)$ increases); too frequent a repair means more fixed payments of $m$.
- The relationship between tests (partially and fully overlapping) and its impact on optimal testing strategies. As summarized in Table 4, the optimal number and fidelity of tests is affected by this relationship: Fully overlapping tests result in fewer high-fidelity tests as opposed to partial overlapping which results in many, lower-fidelity tests as a prescription for effective testing. In most cases, however, the ETF formulation developed in this paper provides a good estimate for the optimal number of tests to be carried out.

The attractiveness of partially overlapping tests arises because they can take advantage of the bargain cost of low-fidelity tests. Fully overlapping tests, by contrast, require increasing fidelity tests to be effective. However, if there is a high fixed cost component of the test, the optimal solution even in the partially overlapping case quickly reverts to our Economic Testing Frequency $\sqrt{dv/2m}$. Our investigation was prompted, in part, by empirical observations of “daily tests,” a practice that seemed to contradict the commonly observed development practice of a few
“big” tests. Our findings show that very frequent testing is economical only if both of the following are true:

(i) the tests are partially overlapping;
(ii) the fixed cost of each test is low.

While the second of these conditions may be evident, we think that the first is not. Frequent testing also can be observed in areas when \( m \) has decreased dramatically, increasingly the case as companies are shifting from physical prototype testing to three-dimensional computer-aided design (CAD). Interference problems—problems where physical parts interfere in geometric space—can now be tested for automatically at a cost that is orders of magnitude less than conventional prototype testing. Even if the tests are fully overlapping, a \( 10^6 \) cost reduction still results in a one-thousand-fold increase in testing frequency (everything else equal); if the tests are partially overlapping, the increase in testing could be much greater.

As a practical example, consider the two examples presented in §3.3 on testing electronic circuits (technical uncertainty) and testing software prototypes with customers (uncertainty about customer needs) once more. For the first example, our findings would suggest that the circuit should be tested very frequently with randomly selected, small families of input patterns (partially overlapping: lower fidelity, high frequency). The circuit prototype itself, however, should be of high fidelity as early as possible (fully overlapping: high fidelity, lower ETF). Similarly, for the software prototype used to get customer feedback, software prototypes should be shown to small focus groups frequently (unless there is a significant fixed set-up cost for each focus group) whereas high-fidelity software prototypes

### Table 4

**Summary of Analytical Results (See Results 1–5 in Section 4)**

<table>
<thead>
<tr>
<th>No. of Tests</th>
<th>Fidelity</th>
<th>Testing Strategy</th>
<th>Assumptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single</td>
<td>Full</td>
<td>The optimal timing is about halfway through the project or after about half of the total problems have arisen</td>
<td>Problems ( t(t) ) increases as ( t ). Rework cost increases as ( dt ). Cost of full-fidelity test is ( m ).</td>
</tr>
<tr>
<td>Single</td>
<td>Variable</td>
<td>The optimal timing remains about halfway through the project. The option of a low-fidelity test is relevant only if ( m &gt; d \sqrt[1/2]{m} ), that is, if the cost of a test is more than 25% of the avoidable cost.</td>
<td>Same assumptions as above. Cost of test of variable-fidelity test = ( mf ).</td>
</tr>
<tr>
<td>Multiple</td>
<td>Full</td>
<td>For multiple full-fidelity tests, the optimal number of tests is ( n = \sqrt[1/2]{m} ), the Economic Test Frequency or ETF, which is the square root of the ratio of avoidable redesign cost and the cost of testing.</td>
<td>Same assumptions as above. Full-fidelity tests are carried at equal intervals ( (t = \gamma, 2\gamma, ..., n\gamma) ).</td>
</tr>
<tr>
<td>Multiple</td>
<td>Variable</td>
<td>Fully Overlapping:</td>
<td>Same assumptions as above. Increasing fidelity ( (f_1, f_2, ..., f_n) = 1 ).</td>
</tr>
<tr>
<td>Multiple</td>
<td>Variable</td>
<td>Partially Overlapping:</td>
<td>Same assumptions as above. No fixed cost per test.</td>
</tr>
</tbody>
</table>

---

1 See section 4.1 for a list and discussion of all model assumptions.
should be used as early as possible and improved at a
frequency of $\sqrt{dv/2m}$.

With the help of the model, one can also address the
decision problem posed in our introduction. Suppose
a manager has a budget of $100,000 and is uncer-
tain whether she should spend it on building higher-
fidelity prototypes or carry out more frequent tests—a
difficult decision that is very common in development
practice. Based on the results of our model (within
a single test method), the decision-making process
needs to include the correlation between tests and the
fixed cost of each test as important variables. If tests
are fully overlapping, the manager will get more value
from her budget if the money is spent on increasing
the fidelity of a few prototypes. If very high-fidelity
tests are infeasible during the early phases (con-
strained fidelity), she should simply instruct develop-
ers to test at the highest fidelity possible. In contrast,
if tests are partially overlapping and have a very low
fixed cost, she would be better off if the money is spent
on more frequent testing.

Optimal testing is an important factor in the effec-
tiveness of product development practice. We there-
fore propose that further studies on this topic may be
of interest to both product development researchers
and practitioners. For example, our paper has not
looked at testing strategies as a function of design
structure. Intuitively, we would expect that optimal
testing varies going from integral to modular struc-
tures, or vice versa. Indeed, Baldwin and Clark (2000)
have modeled the impact of test cost on the value
of modular design structure and found it to be very
significant. They determined that for system-level
testing, the value of product modularity is domi-
nated by the high cost of combinatorial testing. They
conclude that unless testing strategies switch from
system-level to module-level tests (which, of course,
requires detailed knowledge of a module’s contribu-
tion to system-level performance), firms will be inhib-
ited from investing in the development of modular
design. More empirical research on the link between
testing and design structure holds great promise as an
important contribution to product development the-
ory and practice.

As another example of further research, consider
that our paper has been limited to the link between
testing and uncertainty reduction but there is evidence
that test activities also have an impact on communi-
cation and coordination within development teams.
In studying the practice of prototyping, researchers
have found that prototypes often enhance communi-
cation between and within development teams and
can be instrumental in shaping a firm’s culture
(Leonard-Barton 1995, Schrage 1993). As interest in
formal models of communication practices and their
impact on important development variables such as
task concurrency and time continues to grow (Loch
and Terwiesch 1998), it may be interesting to examine
further how testing (and the related prototyping activ-
ities) impacts communication and coordination within
firms. Finally, it would also be useful to examine
optimal testing strategies under conditions of rapid
change. For example, our model could be expanded to
include the ratio of initial uncertainty ($v_i$) and uncer-
tainty that is introduced during product development
($v_f$). In rapidly changing environments, the ratio of
$v_i$ and $v_f$ tends to be larger than in environments
characterized by slow change. As a result, testing
strategies—like other important business decisions—
would be contingent upon the volatility and rate of
change of factors beyond the control of today’s devel-
opment managers.

Appendix Explanation of Result 4a
The three pieces of the cost function are continuous,
that is, they agree at $n = n^*$ and $n = (n^*)^2$. To find the
minimum only requires that we find the minimum of
the middle section, or

$$n + n^* - \frac{1}{3} \left( \frac{n^*}{n} \right)^2 - \frac{1}{6} \left( \frac{n^*}{n} \right)^3.$$

The slope of this function is negative at $n = n^*$ and
positive at $n = n^* + \frac{1}{2}$ (assuming $n^* \geq 2$), indicating a
minimum somewhere in the range $(n^*, n^* + \frac{1}{2})$. It is
also the case that the function is lower at $n = n^*$ than it
is at $n = n^* + 1$, so that if $n^*$ is integral, then it is a
better solution than $n^* + 1$. It is also possible to show
that an integral solution can be no larger than $n^* + 0.6$
or lower than $n^* - 0.4$, meaning that the integral solu-
tion can be no higher than $n^*$ rounded up. While the
precise solution varies with $n^*$, the solution is given
approximately by rounding up or down $n^* + 0.1$ to the
nearest integer; that is, there is a slight bias to rounding up, rather than down.

Acknowledgments
The authors would like to thank our colleagues and four anonymous reviewers for feedback and discussions on earlier versions of this paper. This research was made possible through the generous support of the Harvard Business School Division of Research.

References


Accepted by Luk Van Wassenhove; received November 9, 1998. This paper was with the authors 4 months for 3 revisions.