44

Role of Taguchi Methods in Design for Six Sigma

44.1. Introduction

Quality engineering (also called Taguchi methods) has enjoyed legendary successes in many Japanese corporations over the past five decades and dramatic successes in a growing number of U.S. corporations over the past two decades. In 1999, the American Supplier Institute Consulting Group [1] introduced the IDDOV formulation of DFSS that builds on Taguchi methods to create an even stronger methodology that is helping a growing number of corporations deliver dramatically stronger products.

Taguchi's Quality Engineering Handbook. Genichi Taguchi, Subir Chowdhury and Yuin Wu

Quality engineering pursues both off-line and on-line quality. *Off-line quality engineering* (also called *robust engineering*) provides the core methodologies upon which the first four phases of IDDOV are built: identify project, define requirements, develop concept, and optimize design. On-line quality engineering provides some unique and powerful methods for establishing and maintaining the quality of manufacturing processes within the verify and launch phase of IDDOV.

Since the total set of robust engineering [2,3] methodologies is fully incorporated within IDDOV, robust engineering case studies double as IDDOV case studies. However, the converse does not necessarily hold since IDDOV contains elements that are not encompassed within robust engineering.

Thousands of robust engineering case studies on products ranging from automobiles to medical equipment and consumer electronics produced by U.S. manufacturing corporations have reported, on average, a gain in the signal-to-noise ratio of 6 dB. In terms of product performance metrics such as failure rate, a gain of 6 dB translates into a factor of 2 to 4 (200 to 400%) reduction in failure rate and in warranty cost. A sample of 16 case studies with an average gain of 7 dB is provided in the book *Robust Engineering: Learn How to Boost Quality While Reducing Cost and Time to Market* by Taguchi et al. [2]. These 16 case studies cover a broad range of applications, including acceleration of the growth rate of bean sprouts; improving electronic warfare and communications algorithms; and improving the performance of mechanical, electrical, and electronic systems and components in many different industries. Yuin Wu and Alan Wu provide a "deep dive" into the subject in their book, *Taguchi Methods for Robust Design.* [3]. Subir Chowdhury provides an exceptionally readable introduction to robust engineering and the whole of DFSS in his book, *Design for Six Sigma: The Revolutionary Process for Achieving Extraordinary Profits* [1].

Any manufacturing corporation in any country can rapidly enjoy significant gains by pervasively implementing *design for six sigma* (DFSS), founded on robust engineering. Just as six sigma has helped many corporations enjoy unprecedented financial gains through improvements in their processes, DFSS has helped a growing number of corporations achieve significant, additional financial gains due to improvements in engineered products delivered to customers and design or redesign of business, engineering, manufacturing, and service processes.

The next five sections explore in sequence:

- ❑ Taguchi methods as the foundation of DFSS
- ❑ Overview of DFSS (IDDOV)
- ❑ Overview of quality engineering
- ❑ Linkage between quality engineering and IDDOV
- ❑ Aligning the NPD process with DFSS
- ❑ Technical benefits of IDDOV
- ❑ Financial benefits of IDDOV
- ❑ Taming the intersection of six sigma and robust engineering

44.2. Taguchi Methods as the Foundation of DFSS

Taguchi methods, design for six sigma, and *new product development* (NPD) processes work together. NPD leverages DFSS, which in turn leverages Taguchi methods to achieve competitive leadership. NPD guides a corporation's product development activities. Design for six sigma (DFSS) is carefully designed to support a corporation's new product development process. The three methodologies form a natural hierarchy: (1) new product development (NPD) process, (2) design for six sigma (IDDOV), and (3) Taguchi methods (quality engineering).

The relationship between Taguchi methods and design for six sigma is developed to set the stage for developing the relationship between DFSS and a typical new product development process. A revised new product development process is presented that leverages the unique strengths of DFSS and Taguchi methods.

Overview of DFSS (IDDOV)

American Supplier Institute Consulting Group's formulation of DFSS [1] is structured into five phases:

1. *Identify project.* Select project, refine its scope, develop project plan, and form high-powered team.
2. *Define requirements.* Understand customer requirements and translate them into technical requirements using quality function deployment.
3. *Develop concept.* Generate concept alternatives using Pugh concept generation and TRIZ and select the best concept using Pugh concept selection methods; conduct FMEA [1]. The develop concept may be conducted at several levels, starting with the system architecture for the entire product. Then concepts are developed for the various system elements as needed.
4. *Optimize design.* Optimize technology set and concept design using robust optimization, parameter design, and tolerance design. Optimization is conducted at the system element levels.
5. *Verify and launch.* Finalize manufacturing process design using Taguchi's online quality engineering, conduct prototype cycle and pilot run, ramp-up to full production, and launch product.

The first two phases, identify project and define requirements, focus on getting the right product. The last two phases, optimize design and verify and launch, focus on getting the product right.

The middle phase, develop concept, is the bridge between getting the right product and getting the product right. Bridging across external customer requirements and internal technical requirements makes development of conceptual designs a difficult and critical step in the product development process. As the bridge between upstream and downstream requirements, conceptual designs should creatively respond to upstream requirements developed through the quality function deployment (QFD) process and downstream engineering, manufacturing, and service requirements that do not necessarily flow out of QFD. The central technical requirement that the selected concept and technology set should optimize well to provide good robustness (high signal-to-noise ratio) leads to the potential loopback from optimization to concept development indicated below.

The five phases, IDDOV, are segmented into 20 steps.

Identify Project

1. Refine charter and scope.
2. Develop project plan (plan, do, check, act).
3. Form high-powered team.

Define Requirements (*QFD*)

4. Understand customer requirements (QFD phase I).
5. Build house of quality (QFD phase I).

Develop Concept (*Pugh, TRIZ, and FMEA*)

6. Generate concepts:
    a. Pugh concept generation and creativity toolkit.
    b. TRIZ (theory of inventive problem solving).
7. Select concept (Pugh concept selection process).
    a. Conduct first run of evaluation matrix.
    b. Conduct confirmation run.
    c. Conduct controlled convergence.
8. Conduct FMEA.

Optimize Design (*Taguchi methods*)

9. Develop design planning matrix (QFD phase II).
10. Optimize (concept) design (step 1 of two-step optimization).
11. Adjust to target (step 2 of two-step optimization).
12. Conduct tolerance design.
13. Develop process planning matrix (QFD phase III).
14. Optimize process design.

Verify and Launch

15. Develop operations planning matrix (QFD Phase IV).
16. Finalize operations and service processes.
17. Conduct prototype cycle.
18. Conduct pilot run.
19. Launch, ramp-up, and confirm full production.
20. Track and improve field performance.

Stuart Pugh [4], creator of Pugh concept generation and selection methods, emphasizes the importance of concept design with the statement: “The wrong choice of concept in a given design situation can rarely, if ever, be recouped by brilliant detailed design.” The ability to evaluate robustness of concept designs/technology sets provides the means to avoid the “wrong choice of concept.”

The loopback from optimize design to the develop concept phase can become a critical iteration since a concept design that does not optimize well cannot be recovered later in the process. The only feasible solution is to create a new concept that does optimize well. This is a strong and troublesome statement. The prospect of discarding a concept is far easier said than done. Whether the concept is newly created or an existing product in the field, it is very difficult to recognize and act on the recognition that a concept or embedded technology set or both should be discarded. A concept/technology set represents a significant investment in time and resources that nearly always appears easier to fix than to replace, especially in the absence of a definitive way to differentiate good from bad. The notion of robustness provides the conclusive technical differentiator between good and poor concepts.

The 20-step process outlined above is used for creating new products and processes. The develop concept and optimize design phases combine to provide powerful find and fix firefighting methods and tools. Many of the case studies that contribute to the average gain of 6 dB involve improvement of existing concepts.

IDDOV is

- ❑ An engineering methodology that supports a new product development process
- ❑ Based on the world's best practices, many of which were not introduced into the Western world until the 1980s and are not yet adequately included in university engineering curricula
- ❑ An important implementation of Taguchi's Quality Engineering
- ❑ An engineering process developed by engineers rather than a statistically based process like that commonly employed in Six Sigma

IDDOV emphasizes relatively new methods and tools such as quality function deployment, Pugh concept development, TRIZ, and robust engineering, as described by Subir Chowdhury [1]. These newer, powerful methodologies are relatively familiar within technical communities; however, they are not implemented consistently within Western world NPD processes.

IDDOV is *not* a completely new product development process (NPD process). It is designed as an augmentation to an NPD process. As an augmentation, IDDOV does not encompass all of the elements of mechanical, electrical, chemical, software, and other engineering disciplines necessary to deliver products. IDDOV does not even contain a phase that relates to detailed product and process design of NPD.

Overview of Quality Engineering

Quality engineering (called Taguchi methods in the United States) consists of off-line and on-line quality engineering. (Off-line quality engineering is called robust engineering in the West.)

The purpose of robust engineering is to complement traditional engineering methods with robust methods to increase the competitiveness of new products by reducing their cost and improving their quality, starting with research and development prior to specific product applications. A central focus of robust engineering is robust optimization of conceptual designs and technology sets.

Quality engineering is applicable to all aspects of R&D, product development, and manufacturing process development: *system/concept design, parameter design, and tolerance design*. The three phases are common for technology, product, and manufacturing applications.

Off-line quality engineering (robust engineering) encompasses technology development and product design, which is characterized in more detail.

Technology Development (*R&D Prior to Knowledge about Requirements*)

- ❑ *System/concept design.* Select the best technology set from all possible technology alternatives that might be able to perform the objective function.
- ❑ *Parameter design.* Determine the optimal values of the parameters that affect the performance (*robustness*) of the technology set selected.
- ❑ *Tolerance design.* Find the optimal trade-off between the cost of quality loss due to variation of the objective functions and the cost of technology set materials and processes.

Product Design (*Creation and Optimization of Concepts/Technology Sets*)

- *System/concept design. Select the best system/concept design and technology set from all possible alternatives that can perform the objective function.* System/concept design retains the common meaning of developing innovative conceptual designs that respond to customer, company, and regulatory requirements. Concept design is conducted at various levels in a complex product. At the highest level, system design is the process of partitioning the product into an architectural hierarchy of system elements typically identified by terms such as subsystems, modules, components, and parts. Software implementations involve a corresponding hierarchy of levels. Once system architecture is defined, a conceptual design and technology set for each system element is developed. System design and concept design are frequently used interchangeably, since both terms refer to the process of stratifying a larger entity into its constituent elements.
- *Parameter design. Determine the optimal values of the parameters that affect the performance (robustness and target) of the selected concept/technology set.* Parameter design is the process of identifying and determining the values of design parameters that minimize the sensitivity of the design to sources of variation (i.e., sources of problems, including environmental conditions, customer usage conditions, wear of parts, manufacturing variations, etc.). Parameter design (*robust optimization*) is normally conducted on lower-level system elements that perform a single primary function. Two-step optimization facilitates optimization of concept designs and technology sets. Step 1 maximizes the functional robustness of the product or process concept design. Step 2 adjusts the product or process concept to target without affecting robustness significantly.
- *Tolerance design. Find the optimal trade-off between the cost of quality loss due to variation of the objective functions and the cost of components and technology set materials.* Tolerance design is the process of balancing product cost and quality losses (using the quality loss function). Application of the process is often biased to minimize product cost without significantly sacrificing the cost related to the customer's perspective of product performance, quality, or reliability. The gains realized through robust optimization typically exceed customer needs. Tolerance design identifies elements of the product that can be implemented with lower-cost materials, components, and parts without significantly affecting product performance, quality, or reliability. Tolerance design also works the other way around when the need is to upgrade product elements to reduce field cost.

Tolerance specification depends on the information developed during tolerance design. Tolerance specification is the process of deciding on the tolerance limits that will be entered on drawings and used for quality control of manufacturing and suppliers. Tolerance specification is typically combined with tolerance design.

On-line quality engineering addresses design and quality management of the manufacturing process.

Manufacturing Process Design

- *System/concept design.* Select the best production processes from all possible alternatives.

- *Parameter design.* Determine the optimal values of the parameters that affect the performance (*robustness and target*) of the production processes selected.
- *Tolerance design.* Determine tolerance specifications for the parameters that affect the quality and cost of the production processes.

Quality Management of the Manufacturing Process

Taguchi's on-line quality engineering differs from the typical American process of controlling the manufacturing process against tolerance specification limits. Taguchi's methodology focuses on maintaining the process close to target rather than within control limits. Taguchi observes [5]: "Right now, many American industries use very sophisticated equipment to measure the objective characteristics of their products. However, because some quality managers of these companies do not interrupt the production process as long as their products are within control limits, the objective characteristics of their products are usually uniformly distributed, not normally distributed."

It is clear from studies and common sense that products with all parameters set close to targets are more robust under actual customer usage conditions than products with some portion of parameters set close to control limits. Characteristics set close to control limits are sometimes suggestively called *latent defects* since they can rapidly become very real defects (*or premature failures*) under customer usage.

Linkage between Quality Engineering and IDDOV

The American Supplier Institute Consulting Group (ASI CG) structure of DFSS expands the system/concept design phase of off-line quality engineering into three phases: identify project, define requirements, and develop concept. Parameter design and tolerance design are aggregated under "optimize design." On-line quality engineering pertains to the verify and launch phase of IDDOV. Off-line quality engineering pertains to the following areas:

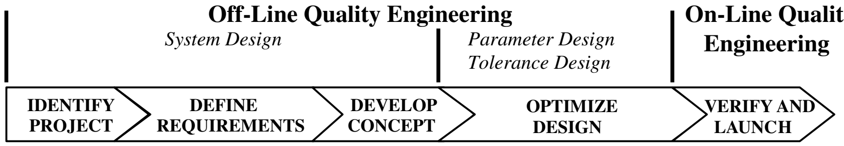
- *System/concept design.* Identify project, define requirements and concept.
- *Parameter design.* Optimize design.
- *Tolerance design.* Perform on-line quality engineering, verify and launch.

The correlation among the three phases of quality engineering and the five phases of DFSS suggest that IDDOV can be regarded as an implementation of quality engineering. The relationship is shown in Figure 44.1.

44.3. Aligning the NPD Process with DFSS

While IDDOV supports rather than replaces a new product development process, it does introduce new engineering capabilities that suggest modifications to the NPD process. As might be expected from the emphasis on robustness above, the

Figure 44.1
Aligning QE and DFSS



refined process will be identified as the new robust product development process (NRPD process).

Some of the problems addressed by the new process are illustrated in the style of a case study that is actually an amalgamation of case studies conducted during the 1990s and early 2000s in several large manufacturing corporations. The products ranged across automotive, office equipment, computer, medical diagnostic equipment, and chemical industries. The synthesized case study sets the stage for developing the NRPD process.

Synthesized Case Study

This study is derived from experience with several large corporations. Each study of an existing NPD process was conducted as background for improving the corporation's capability to deliver winning products consistently. The NPD processes in the study tended to focus more on management of the development process than on management of the engineering process. The NPD processes were universally designed to coordinate the myriad of details related to the development of complex products guided by a plan that contained the types of information identified in the reporter's credo of 5W's1H: who, what, when, where, why, how. The process starts with *why* the product is needed for the business as described in a business case document. The remaining four W's address information about *what* needs to be done *where* by *when* and by *whom. Where* identifies physical locations, organizations, and suppliers. *When* identifies inch-stones and milestones in the plan. *How* refers to the engineering activities, the weakest element in the process.

Managers and engineers, through an elegant software program, could readily access any element of the NPD process. Drop-down menus identified the engineering methodologies and tools that should be used at any point in the process. The 5W's1H and "go/kill phase gate" criteria and process were also available in drop-down menus. A red–yellow–green traffic light evaluation was used on all projects within the program and accumulated up to indicate the status of the total program.

Initial management presentations explaining their in-place NPD processes seem very orderly and, in some cases, very impressive. However, interviews and focus group–style discussions revealed a surprisingly high level of chaos in the engineering process. A seemingly endless series of build–test–fix cycles were conducted throughout the development process–build and test to see if the product met requirements and specifications, and fix the shortfalls, over and over and over.

Engineers and managers alike identified the popular phase gate model that defines every phase gate as a go/kill decision [6] as a major source of problems. If the model (*hardware and/or simulation*) designed and constructed for a build–test–fix cycle in a particular phase of NDP did not meet specifications to the degree required by the phase gate criteria, the team members feared that the program would be killed during the next phase gate review. The continuous series of go/kill tests throughout the development process put the kind of pressure on the team that destroyed confidence and fostered "look good" rather than "do good" behaviors. For example, the teams often struggled to appear as if they were meeting requirements and specifications just before each phase gate review. What better way to look good than to build and show off a sophisticated-looking model that appeared to meet specifications. The new demonstration model was often wasted effort, since its purpose was to help the team to get through the Phase Gate rather than to advance the engineering process.

One senior manager proudly summed up the situation as follows: "Ninety percent of the work gets done in the last 10 percent of the time before the next phase gate review." Others appeared to share his perspective that phase gates were effective fear factors. Whether in engineering or manufacturing, team members "tinkered" until they adjusted parameters to meet specifications, often just barely within specification limits.

In engineering, the spec may be the target value and allowable variation of functional performance that meets customer requirements. Tinkering to meet specs involved changing values of a selected design parameter and one more build–test–fix cycle. In manufacturing, the spec is typically the plus–minus tolerance specification of physical dimensions. Tinkering to meet specs involved adjustment, rework, scrap, and one more build–test–fix cycle.

Generally, manufacturing engineers seemed to understand better than the product engineers that the process of squeezing parameter values just inside "goalpost" specifications led to fragile products. However, their job was defined as meeting specifications, not striving for the centerline target. The conflict between their job description and their knowledge about what was right was a source of deep but silent frustration.

The engineering team members did not share a similar understanding about "just within specs." Their constant daily work was build–test–fix various levels of prototype models in an effort to meet requirements and specifications. That was literally the definition of their job, what they had spent their entire careers doing. The product engineers revealed no recognition that just within specs was even more damaging in engineering than in manufacturing. Anywhere inside the specified limits of allowable variation was regarded as an acceptable target without regard to the obvious fact that expected variation around an off-center target would quickly migrate beyond specification limits.

Neither managers nor engineers were aware of studies (and common sense) that make it clear that products with all parameters set close to targets are more robust under actual customer usage conditions than products with some portion of parameters set close to specification limits, whether in engineering or manufacturing.

Optimization of designs was regarded as just one of the many engineering activities conducted in the later detailed product and process design phases. Different teams within the same corporation conducted optimization in different ways, if at all. Optimization was lightly regarded as "a good thing to do if time allowed."

The degree of chaos was very high in the repetitious build–test–fix efforts to meet specifications under the pressure of go/kill style phase gates. Over half of the product engineers had never enjoyed the opportunity to launch a product into the market. Employee satisfaction, morale, and loyalty were trashed. A bunker mentality had developed as concerns about personal survival overwhelmed the desire to satisfy customers.

Conclusions

The engineering activities within NPD are strongly influenced by the specifics of the engineering methodologies that are used. Most in-place NPD processes do not encompass the newer methodologies, such as robust engineering or IDDOV.

Developing Linkages between NPD and IDDOV

Product development involves two very different activities: (1) concept development and (2) detailed product and process design. These activities are usually further partitioned into four to six phases linked by Phase Gates, as shown in Figure 44.2. The new product development (NPD) process shown in the figure is representative of the many variants. The phase–phase gate construct is a broadly accepted model for NPD [6].

A new product development process is effectively characterized through four elements.

1. *Phase–phase gate structure:* high-level structure, as illustrated in Figure 46.2
2. *Phase processes:* management and engineering activities within each phase
3. *Phase gate process:* activities supporting transitions between phases
4. *Project review process:* steps and style of project review process

PHASE–PHASE GATE STRUCTURE

The relationship of IDDOV and a typical NDP process is indicated in Figure 44.3. The dashed lines indicate the relationship of IDDOV and NPD phases. The stars represent phase gates. The phase gate indicated by the double star and solid line delineates completion of concept development activities and the beginning of detailed product and process design activities. As mentioned previously, the first four phases of DFSS are crowded into preconcept and concept phases of NPD.

While IDDOV can improve the capability to deliver competitive products significantly using the existing NPD process, the alignment with IDDOV is at best awkward. The new engineering methodologies contained within IDDOV suggest changes to the phase–phase gate structure of development processes.

Popular NPD processes simply do not capture the benefits that derive from the relatively new capability to optimize concept designs and technology sets. This important new capability provides competitive benefits that must be captured to enjoy leadership. NPD needs to be modified to capture the benefits of contemporary best practice engineering methodologies. Progressive corporations that refined their NPD process to capture the benefits of IDDOV over the last several years are benefiting from improved competitiveness.

Two-step optimization creates the opportunity to conduct robust optimization (step 1) during concept development and then set the system to target (step 2) during detailed product and process design. Taguchi introduced the notion of optimizing concepts and technology sets in the early 1990s [6]. He states (p. 90): “Research done in research departments should be applied to actual mass production processes and should take into account all kinds of operating conditions. Therefore, research departments should focus on technology that is related to the functional robustness of the processes and products, and not just to meeting customers’ requirements. Bell Laboratories called this type of research ‘two-step design methods.’ In two-step design methods, the functional robustness of product

Figure 44.2
Phases of a typical new product development process

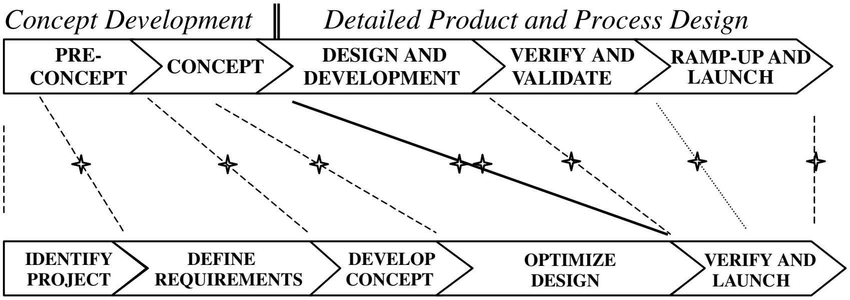


Figure 44.3
Linkage between IDDOV and NPD

and processes is first maximized and then these product and processes are tuned to meet customers' requirements."

This quote correctly suggests that two-step optimization can be brought all the way forward to R&D activities that precede NPD concept development activities. A refined new product development process explicitly identifies optimization upfront in the development process. Figure 44.4 depicts one possible structure for a refined NPD process supported by IDDOV that will serve as the foundation for building a winning development process.

Elevating optimization to appear explicitly in what is now properly identified as the new robust product development (NRPD) process is an important refinement. Optimization serves somewhat different purposes in its two appearances in the NRPD phases. Some subsystems may involve new concept designs and/or new technology sets, while other subsystems may involve carryover designs from precedent products.

New concept designs should be optimized (step 1 of the two-step optimization process) in the concept optimization phase and adjusted to target (step 2) either in the concept optimization phase or the optimization, design, and development phase.

Candidate carryover designs should be assessed for potential robustness prior to accepting them as elements of the concept design. Most carryover designs can be improved by conducting robust optimization experiments. Such carryover designs might be both optimized and set to target in the optimization, design, and development phase.

The differences between NPD and NRPD depicted in Figures 44.3 and 44.4, respectively, have huge implications. As noted earlier, optimization is normally re-

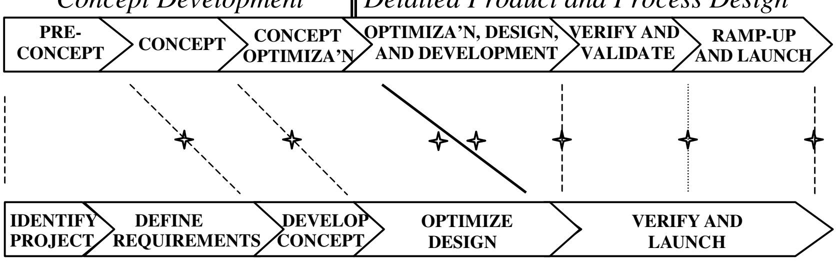


Figure 44.4
Optimization cited explicitly in NRPD

garded as just one of many functions performed during detailed design that are too numerous to be identified in the high level of NPD phases. Elevating optimization to appear explicitly in the phase structure of NRPD is a big change. Pulling optimization forward to become part of the concept development activities is an even bigger change. The ability to optimize concepts and technology sets early in the development process streamlines the NRPD engineering process to significantly reduce the risk of innovation, enhance engineering excellence, reduce likelihood of canceling a development program, reduce the number of build–test–fix cycles needed, shorten development schedules, and reduce development, product, and warranty costs.

Figures 44.3 and 44.4 provide a reasonable representation of the linkages between IDDOV, NPD, and NRPD, respectively, for relatively simple products or for subsystems within a more complex product. The figures provide a good representation for suppliers of functional subsystems to original equipment manufacturers that deliver complex products. However, a different representation is needed for complex products.

The development of complex products usually involves a number of IDDOV projects in various points within the NRPD process, as depicted in Figure 44.5.

The activities in the verify and launch phase of IDDOV differ between the different applications. In the concept development phases, a lowercase *v* indicates the "conduct confirmation run" step of the robust optimization process. In the detailed design and development phases, *V* references all of the activities necessary to verify and validate product and process design, ramp-up production, and launch the product. In the verify and validate phase of NRPD, the activities indicated by *V*? are situational, as suggested by the question mark. Complexity determines how to align IDDOV projects with the product or subsystem under development.

PHASE PROCESSES

Xerox and numerous other corporations have encompassed off-line quality engineering (robust engineering) in their NPD [7]. In this section, the activities within each of the phases of NRPD are described briefly. The full significance of the NRPD process may not become evident until the compatible phase gate processes are introduced in "Phase Gate Processes."

In the first three phases of the NRPD process (preconcept, concept, concept optimization), the team focuses on getting the right product concept up-front for a flawless GOOD START transition from concept development activities to detailed product and process design. In the three detailed product and process design

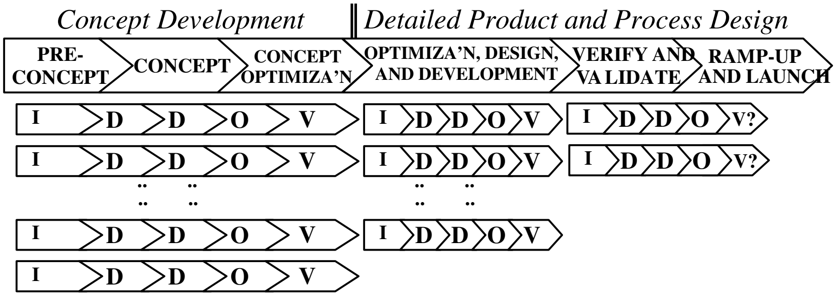


Figure 44.5 IDDOV projects conducted in different phases of NRPD for a complex product

phases (optimization, design, and development; verify and validate; ramp-up and launch), the team focuses on getting the product right.

1. *Preconcept.* Understand customer requirements and market potential, translate customer, company, and regulatory requirements into technical requirements (e.g., QFD), develop product description and business case.
2. *Concept.* Develop product (system) concept design that meets high-level requirements. Develop concepts/technology sets and identify carryover designs for all subsystems in the architectural hierarchy. Develop optimization plans.
3. *Concept optimization.* Two-step optimization dramatically simplifies the new product development process by separating optimization of technology sets and concepts from the challenges of meeting product specifications (Figure 44.6).

The first step of two-step optimization provides the opportunity first to optimize concept/technology set without regard to detailed product specifications. The optimized concept/technology set is subsequently set to product specific specifications in the second step.

1. Optimize the concept/technology set (determine the values of control factors to maximize the SN ratio).
2. Adjust the system to meet the specifications (select the adjustment factor that does not strongly affect the SN ratio and use it to set the system to target specifications).

For new concept designs and technology sets, step 1 is best performed in the concept optimization phase, and step 2 may be conducted in either the concept optimization phase or in the optimization, design, and development phase, depending on circumstances, such as knowledge of target applications. For subsystems, technologies, and concepts carried over from precedent products, optimization might be conducted in either the optimize concept phase or in the optimization, design, and development phase.

If the carryover subsystem has previously been optimized using robust optimization, it may only be necessary to conduct step 2 to set the subsystem to the new product specifications.

1. *Optimization, design.* Conduct detailed design, and optimize designs as needed.
2. *Development.* Conduct modeling, simulation, and bench tests; finalize product and process designs; and release drawings: build and test manufacturing intent prototypes.

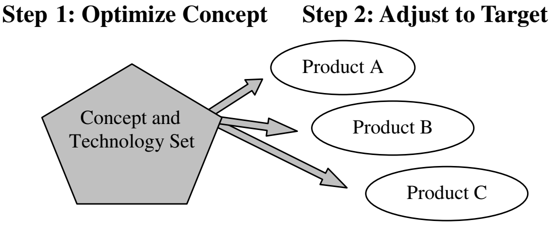


Figure 44.6
Optimized concept and technology set directed to different targets

3. *Verify and validate.* Finalize manufacturing process design including quality control. Taguchi's on-line quality engineering can usefully be applied in conducting this task. Evaluate manufacturing intent prototypes against performance, quality, reliability, and durability requirements under simulated customer usage conditions to verify product function and user friendliness. Conduct a pilot run to validate manufacturing and service processes.
4. *Ramp-up and launch.* Ramp-up production and launch product

PHASE GATE PROCESS

The phase gate and project review processes are closely linked. The project review process is broken out because it can be used at numerous points during the development process beyond the formal phase gates. The phase gate processes need to be designed to be compatible with the engineering and management processes. The ability to establish the technical sturdiness of the combination of a conceptual design and its technology set through robust optimization early in the development process opens up the opportunity to streamline the phase gate processes. The ability to verify the robustness of a concept and technology set decreases the likelihood of unanticipated downstream technical problems dramatically. Hence, the need for demoralizing, downstream go/kill[6]-style phase gates is largely eliminated. Proactive help/go-style phase gates become more effective in the context of early optimization. The intent of help/go-style phase gates is to help teams accelerate progress by resolving issues and demonstrating readiness to move forward into the next phase. Forward-looking preparation for entry into the next phase creates a more positive environment than backward-looking inquisitional evaluations against exit criteria.

The only go/(not-ready-yet or kill) phase gate resides between concept and technology activities and detailed product and process design activities, as indicated in Figures 44.4 and 44.5. In many corporations, the creative advanced development activities are conducted in different organizations by different teams from product development activities. The two-organization model makes the transition from conceptual development activities to detailed product and process design activities a significant event with the commitment of major resources. In the early 1980s, Xerox identified this event as GOOD START.

The primary criteria for a GOOD START are (1) business readiness and (2) technical readiness. *Business readiness* requires that completed business plans project financial attractiveness. *Technical readiness* requires that a strong product concept that meets upstream customer requirements and downstream technical requirements has been selected, and that technical feasibility based on mature technologies has been demonstrated.

The ability to optimize concepts and technology sets prior to transitioning to detailed product and process design supports the effective implementation of the GOOD START discipline. Robust optimization provides solid evidence about the strengths of concept designs and technology sets. This is an enormously important new capability. The engineering and management teams can know whether a new technology or a new concept design will be robust under actual customer usage conditions very early in the development process, before making the major resource commitment necessary to support detailed product and process design.

In addition, robust optimization assessment methods can be used to evaluate potential carryover designs prior to making reuse versus new concept design decisions. Robust optimization helps to remove uncertainties about technical readiness prior to GOOD START.

The GOOD START model features a strong management commitment to take the product to market with only two acceptable "showstoppers": (1) an unanticipated competitive action that makes the product nonfeasible in the marketplace, and (2) technical problems that cannot be recovered within the window of market opportunity.

A disciplined GOOD START with a hard commitment to go to market reduces the likelihood of killing a program for financial convenience, such as trying to fix this quarter's profit problems, cost overruns, staffing shortfalls, and a litany of other excuses.

The GOOD START model promotes continuity of effort, confidence, winning attitudes, and engineering excellence, in contrast to models that treat every phase gate as a go/kill decision. [6] When GOOD START is defined as the only go/(not-ready-yet or kill) phase gate, management and program/project team members are strongly motivated to get the potential program right up-front. The rigorous up-front process required to get through the GOOD START phase gate dramatically increases the probability that the program team will launch an outstanding product successfully.

PROJECT REVIEW PROCESS

A strong PROJECT REVIEW process fosters engineering excellence by ensuring excellence of execution of every step of the development process. A three-step project review process supports excellence of execution: (1) self-inspection, (2) peer (design) review, and (3) management review. The review process helps to ensure that all of the actions necessary to prepare for GOOD START have been executed properly. The project review process continues to ensure excellence of execution throughout the development process.

Team self-assessment should be facilitated by a team member. The facilitator role can be rotated among team members. Team members can assess their own work or ask for other team members to assess their work. The self-assessment should consider:

- ☐ Degree of excellence in carrying out process steps
- ☐ Did team complete their work or run out of time?
- ☐ Things done right and things gone wrong
- ☐ Ideas for improvement
- ☐ Were results world class? (While assessment should focus on execution of process, results, when available, provide a strong indication of excellence of execution. *In-process checks are leading indicators. Results are lagging indicators.*)

Document the assessment as a text document and presentation slides suitable for use as input to the expert peer review team. *Expert peer review (design review)* should be facilitated by an external person if possible. Reviewers should be external, neutral, and competent/expert in the field. The review team usually consists of two to four members. They should be reminded that their job is to help improve the process and the outcomes, not to criticize. The review team should assess the self-assessment, coherence of team, and external factors, including suppliers, customers, and management.

The output of the review should be submitted to the team leader, black belt, or master black belt. Document the assessment as a text document and presentation slides suitable for use as input for the management review.

Management reviews should be motivational and helpful. Makeup of the review committee might include top executives, involved managers, champions, and, when appropriate, supplier and customer representatives (i.e., the extended team).

Chair management reviews with a supportive style:

☐ Avoid inquisitional, backward-looking questions.

☐ Ask “How can I help?”, forward-looking questions.

☐ Ask what, not who.

☐ Be a friend, not a command-and-control general.

☐ Drive out fear. *Fear stifles clear thinking, innovation, and bold actions.*

☐ Knock down barriers, remove obstacles, solve problems.

☐ Recognize mistakes as progress. *Suggest, only partially in jest: “Make mistakes as rapidly as possible to get them out of the way as quickly as possible.”*

“Out-of-the-box” suggestions:

☐ Discourage dry runs. *Grant trust to the people, not to the system.*

☐ Encourage “chalk talks” on white boards and flipcharts.

Foster a passion for excellence. Seek to understand the quality of events. Do the team members feel that they were encouraged and given the time to pursue excellence in performing each step of the design process? Did team members perform a self-assessment and a peer review to ensure excellence prior to this management review?

A properly structured NRPD supported by IDDOV, GOOD START, and project review management processes creates a powerful development and engineering environment. Highly motivated teams with winning attitudes use the new processes and methodologies to deliver winning products that help a corporation profitably grow market share and revenue.

44.4. Technical Benefits of IDDOV

The success of quality engineering is legendary in Japan and is recognized increasingly in Western countries. This article was written at a time when six sigma was consuming management attention. Six sigma enjoys enormous success in improving existing processes to yield quick financial benefits. As a relative of six sigma, design for six sigma (IDDOV) is gaining recognition as the means for creating innovative, low-cost, trouble-free products on significantly shorter schedules. In addition, robust optimization, the “O” in IDDOV, is gaining recognition as a powerful problem-solving methodology for developing robust solutions that prevent the reappearance of problems.

The purposes of the three methodologies might be summarized as follows:

☐ Six sigma is an improvement process.

☐ Robust optimization is an improvement and prevention process.

☐ IDDOV is a creation, innovation, improvement, and prevention process.

In *six sigma,* improvement is achieved by digging down (*analyze*) into the details of an existing process or product to find and then eliminate the cause of the problem (*improve*). It is a powerful find-and-fix process that is helping many corporations to achieve huge performance and financial gains. Analyze and improve are two of the phases of the six sigma DMAIC process: define, measure, analyze, improve, control.

A difficulty with six sigma–style problem-solving processes, especially when applied to product problems, is that whacking down one problem frequently causes another problem to pop up. This good–bad dichotomy is somewhat pejoratively characterized as "whack-a-mole engineering." The game consists of a number of holes in a board with a mole's head sticking up through one of the holes. When one mole is hit on the head, another mole pops up out of another hole.

A number of actual case studies illustrate the equivalent engineering game. In one study, a team worked for many months to reduce the audible noise of a drive belt. But another problem popped up; the already short life of the belt was further shortened by a factor of 2. In another study, a team reduced the audible noise due to piston slap in a diesel engine. Problems of performance and fuel use popped up.

In *robust optimization,* improvement means something very different from its meaning in six sigma. Rather than seeking to eliminate an identified problem such as the audible noise in the aforementioned drive belt, robust optimization seeks to maximize the performance of the intended function of the product or process system. In the drive belt case study, robust optimization simultaneously eliminated the audible noise and doubled the life of the belt [7]. In the diesel engine example, optimizing engine efficiency dramatically reduced audible noise while improving performance and fuel use.

A simple explanation for working on the intended function rather than the problems (e.g., audible noise) has proved useful in helping nonexperts begin to accept the principle of signal-to-noise ratio based on energy thinking. Think of functions (intended and unintended) as energy transformations. By conservation of energy, maximizing the energy in the intended function (signal) minimizes the energy available (noise) to cause all unintended functions (all problems), not just audible noise. Hence, symptomatic problems such as inefficiency, noise, vibration, aging, wear, and heat are all improved by one set of robust optimization experiments to maximize signal-to-noise ratio. The tedious task of digging to find root causes of a symptomatic problem is bypassed.

Old AM radios provide a familiar example of signal and noise contributions to performance. Signal is the intended music. Static is the unintended noise. The total energy received by the radio is the sum of the energy that produces music, the signal, and the energy that produces static, the noise. Increasing the portion of total energy that produces music clearly reduces the portion of the total energy that is available to produce static. In an ideal radio, the total energy would go into producing music, leaving no energy to cause static. The function of the radio would be ideal. No static would be present to interfere with the music.

In robust engineering, if a function is performed perfectly, it is called the *ideal function.* Of course, the ideal function does not exist in our world. Try as we might, perpetual-motion machines do not exist. However, the measure of how far the value of the actual function differs from the value of the ideal function is a useful measure of robustness. It tells us about the relative volumes of music and static.

The objective of robust optimization is to maximize the relative volumes of music and static. This process moves the actual function as close as possible to the ideal function. The measure of robustness is the *signal-to-noise ratio,* the ratio of the energy in the intended function (signal) to the energy in the unintended functions (noise).

Assume that the ideal system response, y, is a linear relation to an M. Then the ideal function is $y = \beta M$. Now assume that the actual response to M is some function of M together with various system parameters, $x_1, x_2, \ldots, x_n$, then $y = f(M, x_1, x_2, \ldots, x_n)$. The actual function may be rearranged into two parts, the ideal function (useful part) and the deviation from the ideal function (harmful part) by adding and subtracting the ideal function, βM:

$$y = f(M, x_1, x_2, \ldots, x_n) = \overset{\substack{\text{Ideal}\\ \text{Function}}}{\beta M} + \overset{\substack{\text{Deviation from}\\ \text{Ideal Function}}}{[f(M, x_1, x_2, \ldots, x_n) - \beta M]}$$

The ideal function represents all the radio signal energy going into the music with no energy available to cause static. The deviation from ideal is the portion of energy in the actual function that goes into causing static. Robust optimization is a methodology for maximizing the portion of energy that goes into the ideal function, which in turn minimizes the energy available to cause deviation from the ideal function. Referencing the functions above, robust optimization is the process of minimizing the deviation from the ideal function by finding the values of controllable x_i's that move the actual function, $f(M, x_1, x_2, \ldots, x_n)$, as close as possible to the ideal function, βM. When the value of the actual function is close to the value of the ideal function, the system is robust.

Some of the x_i's are usually not controllable, such as environmental and usage conditions, variations in materials and part dimensions, and deterioration factors such wear and aging. Uncontrollable factors are called *noise factors.* When the value of the actual function remains close to the values of the ideal function for all anticipated values of the noise factors, the system is robust in the presence of noise. Such a system is insensitive to sources of variation.

Taguchi uses the terms *useful part* and *harmful part* for ideal function and deviation from ideal function, respectively. The harmful part is the portion of energy that goes into causing problems. These terms provide the rationale for working on the useful part, the intended function of making music. Maximizing the energy in the useful part minimizes the energy available to go into the harmful part. The harmful part can cause multiple problems. Maximizing the actual function minimizes the harmful part that causes problems. Robust optimization is truly both an improvement and a prevention methodology. All moles are kept down permanently with a single whack.

In *IDDOV,* the methodologies for the creation of products and processes are built around the robust optimization methodology to formulate an engineering process that supports new product development processes described previously. In addition to being a creation, improvement, and prevention methodology, the develop concept phase of IDDOV provides the maximum opportunity for innovation.

IDDOV supports NRPD to deliver *"better, faster, cheaper"* benefits, including:

☐ Innovative, low-cost, reliable products that many customers around the globe purchase in preference to competitive offerings

- Low warranty cost and delighted customers
- Low development cost and reduced time to market
- Improved employee satisfaction and loyalty
- Growth of profit, revenue, and market share

The signal-to-noise (SN) ratio is, of course, the single measure of robustness, and gain in the SN ratio is the single measure of improvement. The SN ratio is an abstract measure with little meaning until it is related to *better, faster, cheaper* benefits.

In this section concerning technical benefits, the metrics selected to illustrate the relation between gain in SN to better, faster types of benefits include (1) range of variation and sigma level (better), (2) concept/technology set evaluation (better and faster), (3) failure rate (better), and (4) innovation (better and faster). *Cheaper* benefits are addressed appropriately in the financial benefits section.

Range of Variation and Sigma Level

A pictorial representation illustrating improvement in a manufacturing process achieved by conducting a robust optimization experiment is shown in Figure 44.7. A typical gain of 6 dB is assumed. The reduction in the range of variation depicted is given by the formula in the figure, $\sigma_{OP}/\sigma_{BL} = (\frac{1}{2})^{gain/6}$. When the gain equals 6 dB, the formula reduces to $6\sigma_{OP} = \frac{1}{2}\sigma_{BL}$ (a factor of 2 reduction in the range of variation).

The sigma level is related to the number of standard deviations (σ's) that fit within the control limits. The factor of 2 increase in sigma level from 3.5σ to 7σ is correlated with the increase in SN ratio along the vertical axis. The additional improvement to 8.5σ assumes that a 1.5σ shift existed that was eliminated by optimization.

The initial sigma level of 3.5σ was arbitrarily chosen as roughly representative of current manufacturing industry performance. The corresponding baseline SN ratio of 24 dB is also chosen arbitrarily, for illustrative purposes. For a troublesome

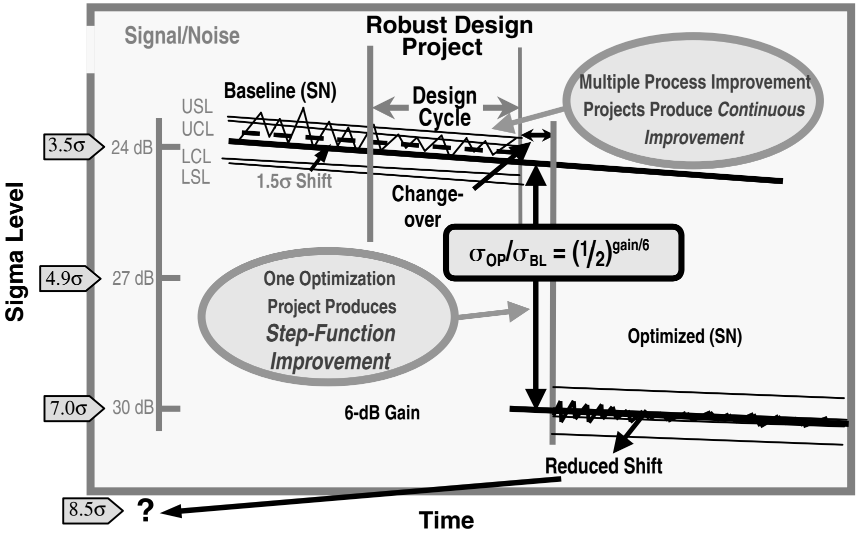


Figure 44.7 Improvement due to robust optimization

element of a system, a more representative choice for the initial sigma level might be 2σ; then the sigma level would increase to 4σ for a gain of 6 dB.

Dealing with the famous 1.5σ shift can be troublesome. Perhaps the shift was not present prior to optimization. Does it apply to distributions of variation in products as well as processes? Six sigma programs often assume that the 1.5σ shift is always present. This assumption does not work in the context of Taguchi methods.

MANUFACTURING PROCESS PERSPECTIVE

Both the similarities and the differences between Figure 44.7 and the Juran trilogy [8] shown in Figure 44.8 are striking. A small difference is the choice of metrics. The vertical axis in the Juran trilogy is the cost of poor quality, which, of course, is related directly to the SN and sigma level (added to figure) used in Figure 44.7. The most striking difference is the slight tilt in the control charts in Figure 44.7 that results from continuous improvement actions. As the range of variation is reduced over time, the sigma level increases to create the slight tilt.

Juran's trilogy contains three steps for improving quality:

1. *Quality planning.* Establish plans to improve quality to meet requirements.
2. *Quality control.* Get process in control before trying to improve it.
3. *Quality improvement.* Make improvement and maintain process control at new levels.

The Juran trilogy represents one action in a sequence of continuous improvement actions. The axis of the control chart remains horizontal between improvement actions. The 1.2σ improvement indicated in Figure 4.8 is normally regarded as a large improvement for a manufacturing process but a small step compared to the gain typically achieved using robust optimization. The tilted lines in Figure 44.7 represent a series of Juran trilogies (i.e., the continuous improvement that results from a sequence of small step-function improvements smoothed into lines).

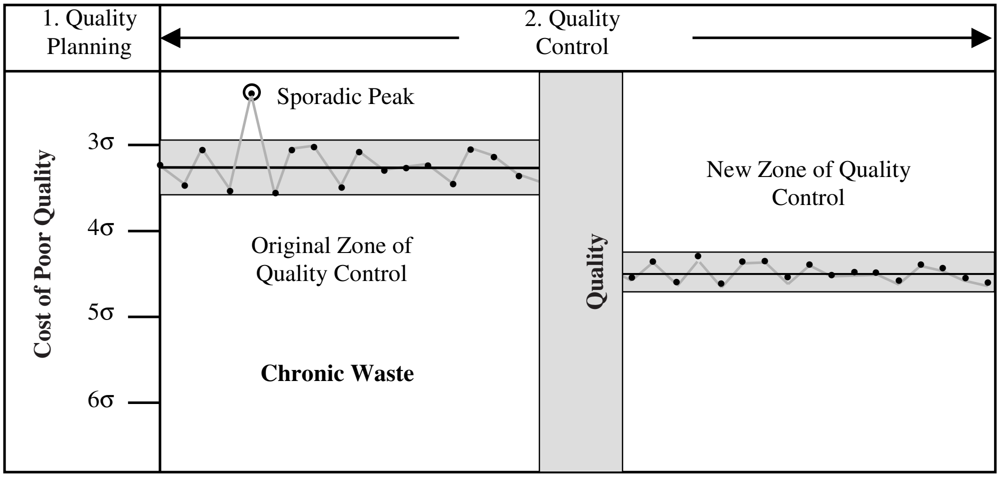


Figure 44.8
Juran's quality trilogy with representative sigma levels added (Adapted from Joseph M. Juran and A. Blanton Godfrey, *Juran's Quality Handbook*, McGraw-Hill, New York, 1999, Section 2.7.)

PRODUCT PERSPECTIVE

The part-to-part variation in a manufacturing process is one of three sources of variation. There are two other sources of variation in a product: internal and external noises. *Internal noise* includes aging, wear, vibration, heat, and so on; *external noise* includes environmental conditions and actual usage profiles.

The before-and-after variation depicted in Figure 44.7 can be regarded as representing a product in the presence of the three sources of variation: part-to-part variations, internal noise, and external noise. The depicted control charts are reinterpreted as tolerance charts that show variations in the context of specification limits (the manufacturing control limits are ignored since they are not appropriate to tolerance charts). Assume that the graphical representation of variation is made with data taken from an instrumented product operated under simulated usage conditions, as many manufacturers do as part of their internal evaluation processes.

The tolerance limits are determined by the degree of variation in performance that causes a user to take corrective action. Two types of failures are identified: functional failure (tow truck to service center) and performance degradation failure (drive to service center).

The slope of the tolerance charts relates to changes in the range of variation over time. The failure rate should change in the same way as range of variation does. The negative slope of the tolerance chart indicates improvement, a positive slope indicates degradation, and a zero slope indicates no change over time.

The objective of robust engineering and IDDOV is to maximize the robustness. Robustness is defined by Taguchi as "the state where the technology, product, or process performance is minimally sensitive to factors causing variability (either in manufacturing or user's environment) and aging at the lowest unit manufacturing cost."

With this definition, the slope of the tolerance charts should remain near zero, or horizontal, in the user's environment, with minimal variation in product performance (well within the upper and lower tolerance limits), as shown for the optimized control chart shown in the *lower* right-hand corner of Figure 44.7.

Concept/Technology Set Evaluation

A good concept/technology set optimizes very well, yielding, on average, a 6-dB gain. A poor choice of concept or technology set will not optimize well. The distinction between "optimizes well" and "does not optimize well" is not a well-defined boundary. Rough guidelines suggest that more than a 2.5-dB gain is good and less than a 1.0-dB gain is bad. *Good* and *bad* refer to the quality of the concept. Judgment is necessary for gains between 1.0 and 2.5 dB, and for that matter, any gain. Relations are readily constructed from the formula, $\sigma_{\mathrm{OP}}/\sigma_{\mathrm{BL}} = (\frac{1}{2})^{\mathrm{gain}/6}$, to help make judgments. The reduction in range of variation versus gain (dB) for $(\frac{1}{2})^{\mathrm{gain}/6}$ is as follows:

Gain (dB):	0.1	1.0	1.5	2.0	2.5	3.0	6.0	10.0	18.0	24.0
Range reduction (%):	1	11	16	21	25	29	50	69	88	94

Range of variation provides a useful, although incomplete visual depiction of the degree of robustness. In Figure 44.9, the vertical goal posts represent upper and lower tolerance limits. The separation between the goalposts, USL − LSL, is the design tolerance around some target $\pm\Delta$, where $2\Delta = \mathrm{USL} - \mathrm{LSL}$. The chart

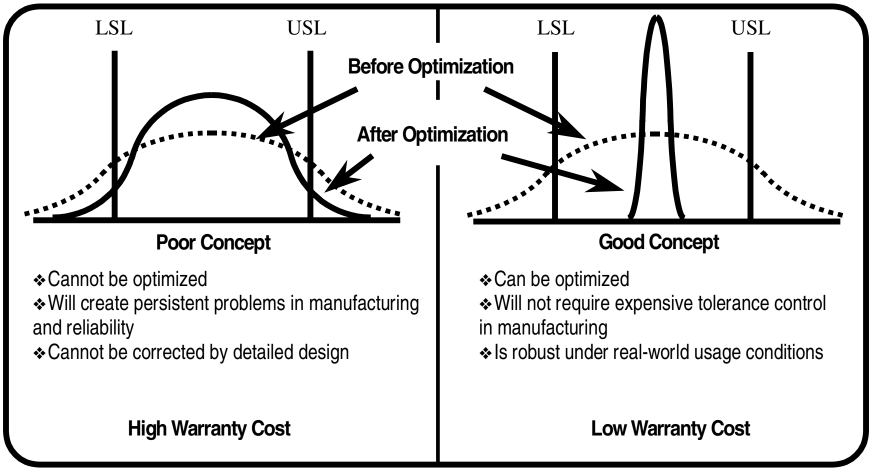


Figure 44.9
Poor concepts cannot be recouped

shows variation compared to tolerance specifications. The distributions illustrated may represent product or process variations.

Robustness is a valuable metric for detecting poor concepts or technology sets. A concept/technology set that does not optimize well is probably a poor choice of either the conceptual design or the technology set.

Poor concepts or technology sets are the sources of chronic problems that reappear program after program to provide permanent employment for expensive armies of firefighters. Since neither poor concepts nor technology sets can be recouped by brilliant detailed design or firefighting, the only solution is to throw out the old design and invest in the development of a new concept or technology set that does optimize well. Throwing out a design is always a difficult decision with potentially major near-term financial consequences.

Most corporate environments, accidentally or intentionally, place higher priority on *urgency* about near-term cost avoidance than on *importance* about longer-term financial benefits. Costs are immediate and certain, while benefits are longer-term and uncertain. Near-term cost pressures to salvage an existing concept or technology set tend to outweigh potential, longer-term financial benefits of reduced warranty cost and improved customer satisfaction. *The urgent always pushes out the important.* The actual costs of fruitless and endless efforts to salvage unsalvageable concepts are usually not very visible.

In addition to plaguing current field performance, bad concepts get carried over into next-generation offerings to plague future field performance. As development schedule and cost pressures increase, so do the pressures to carry over precedent designs into the next development program. Development engineers like to create designs; that is, after all, what they do. In addition, downstream firefighters usually persist with the belief that they can fix the problems with the current concept or technology set; that is, after all, what they do. In the presence of conflicting inputs from well-intended and respected development and firefighting engineers, together with the absence of objective measures about the potential quality of a concept or technology set, managers have little choice beyond doing whatever seems necessary to fix and maintain the current concept and technology set.

Robustness is a conclusive discriminator of the quality of a conceptual design or technology set. If a concept or technology set is not robust and cannot be optimized to be robust, it must be replaced with an alternative concept or technology set. Features and functions can usually be changed to meet changing customer needs. However, factors such as performance, reliability, and cost that determine growth rate, market share, and profitability are strongly related to robustness, which is inherent in the technical attributes of the concept and technology set.

Failure Rate

Taguchi asserts that robust optimization reduces the failure rate (FR) by a factor of $(\frac{1}{2})^{\text{gain}/3}$, leading to the equation, $\text{FR}_{\text{optimized}} = \text{FR}_{\text{initial}} \times (\frac{1}{2})^{\text{gain}/3}$.

Again, assuming an average gain in the SN ratio of 6 dB, the failure rate is reduced by a factor of 4. As with the range of variation, the failure rate of a poor concept may not be improved significantly by optimization (i.e., the GAIN realized is substantially less than 1.5 dB). In Figure 44.10, the traditional bathtub curve is used to provide a familiar graphical representation of the impact of optimizing a good concept design.

Many teams build in a safety factor by using the more conservative gain/6 rather than the gain/3 relation to project improvements in failure rates. In Figure 44.10, the more conservative gain/6 relation is used, which yields a factor of 2 reduction in failure rate rather than the more aggressive gain/3 relation, which would yield a factor of four improvement. Whether to use gain/3 or gain/6 to make failure rate predictions is a matter of judgment.

The $(\frac{1}{2})^{\text{gain}/3}$ or $(\frac{1}{2})^{\text{gain}/6}$ factor applies to the failure rate, not initial quality, reliability, or durability, which are identified in Figure 44.10 with portions of the bathtub curves. Reliability is calculated from the failure rate. Condra [9] defines reliability as "quality over time," a definition that Juran repeats in his handbook [8]. For a constant failure rate typical of electronic systems, reliability, $R(t) = e^{-\text{FR}\cdot t}$.

For failure rates that change over time, $\text{FR}(t)$, typical of mechanical systems and certain electronic components, reliability, $R(t)$, changes with time according to the formula $R(t) = \exp[-\int \text{FR}(t)\,dt]$, where the integral range is 0 to t. Reliability predictions are typically made using the Weibull function.

In either case, changes in the failure rate [normally called the *hazard rate*, $h(t)$, in reliability engineering] are related by an exponential function to changes in reliability. The brief discussion about reliability is presented only for the purpose of clarifying how the failure-rate bathtub curves in Figure 44.10 relate to Taguchi's $(\frac{1}{2})^{\text{gain}/3}$ factor.

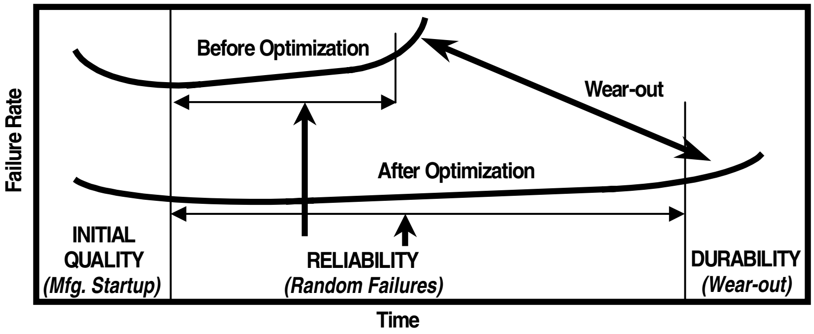


Figure 44.10 Good concepts optimized to deliver improved initial quality, reliability, and durability

Robust optimization of a good concept is the most effective way to improve reliability. Reliability engineering is more about predicting reliability from data collected from measurements of portions of the product than it is about improving reliability. Condra [9] provides an early application of Taguchi methods as the best way to improve reliability in the context of reliability engineering.

Find-and-fix improvement efforts may be constrained by major sunk investments in tooling, and so on, that can severely limit the selection of control factors. If control factors with a major impact on robustness are not available, the amount of improvement may be disappointing. This said, the average gain realized, even with the constraints of "fixing" existing products, appears to be around the average of 6 dB. Perhaps the aforementioned build–test–fix process left too many parameters just inside the goalpost specs so that even constrained optimization experiments yielded significant gains in the SN ratio. Surely even better performance could be achieved by conducting robust optimization experiments during the development process.

Innovation

R&D labs are seldom equipped or funded to fully evaluate and maturate promising innovations to levels suitable for product applications. R&D teams tend to work around these constraints by tying emerging technologies and innovations to product development programs. This high-risk practice contributes to schedule slips and canceled programs. Understandably, after several disappointments, management teams become reticent to take innovations to market. Over time, the rate of innovation slows to a snail's pace.

Robust optimization fosters innovation, a benefit of robust engineering that is seldom heralded. Taguchi methods do not cause the word *innovation* to spring to mind. Nevertheless, the ability to conduct robust optimization experiments on concepts and technology sets in R&D or during the concept phases of NRPD prior to GOOD START significantly reduces the risk of implementing innovations into a development program.

Robust optimization can change the way that R&D is conducted and provide a way out of the "innovation paralysis" trap. [6] The major challenge for most R&D teams is in devising a noise strategy that represents real-world conditions that may not be familiar to the teams. Technology sets and concepts can be optimized efficiently without regard to product specific requirements. If an innovative concept/technology set optimizes well, it can be taken to market with minimal risk. Successes breed confidence, and confidence will again open the spigots of innovation.

Progressive corporations are beginning to move application of robust optimization upstream into R&D laboratories to help accelerate the rate of innovation. With increasing upstream applications, corporations are beginning to take advantage of the full application range of Taguchi methods from research through engineering and manufacturing to products already in the market.

44.5. Financial Benefits of IDDOV

Taguchi methods provide new ways to both reduce cost and make financial projections with confidence. A critical factor propelling the popularity of six sigma is the consistent use of money as the measure of success. The financial impact of every project is calculated. In typical applications of six sigma such as reducing scrap and rework in manufacturing, the calculations are straightforward using cost of quality methods: cost of (poor) internal quality, cost of (poor) external quality, and cost of lost opportunity.

Major impacts of DFSS are derived from product improvements. Relating product design changes to financial benefits can be very difficult. For example, projections of reduction in warranty cost that result from elimination or reduction of root causes depend on the difficult task of relating design changes to reductions in the frequency of occurrence (failure rate) of the problem. In contrast, robust optimization provides an explicit formula that relates gain in signal-to-noise ratio to failure rate, which is readily converted to warranty cost.

The intent of this section is to indicate the types of financial benefits that can be calculated with confidence, with occasional pointers about how to carry out the easier calculations. It is not intended as a complete treatise on making financial projections.

Three types of financial benefits are described: (1) warranty/cost service, (2) product cost, and (3) loss to society (balance between internal and external cost).

Warranty Cost/Service Cost

Calculations of field cost start with the now familiar relation for failure rate. To illustrate a reasonable assumption, the failure rate formula is adjusted by taking an exponential factor between gain/6 and gain/3 (e.g., gain/4.5):

$$\mathrm{FR}_{\text{new}} = \mathrm{FR}_{\text{initial}} \times \left(\tfrac{1}{2}\right)^{\text{gain}/4.5}$$

This relationship provides the basis for making financial projections of service cost.

Service cost is inclusive of warranty cost and customer cost.

Define annual service cost = (no. failures/year) × (average cost per failure)

Then

$$\text{annual service cost}_{\text{new}} = \text{annual service cost}_{\text{initial}} \times \left(\tfrac{1}{2}\right)^{\text{gain}/4.5}$$

Assuming an initial annual service cost of \$500, a gain of 6 dB reduces the annual service cost to \$200:

$$\text{annual service cost}_{\text{new}} = \$500 \times \left(\tfrac{1}{2}\right)^{6/4.5} = \$200$$

This simple means of projecting changes in service cost (warranty and customer cost) provides better estimates than more complex alternative methods.

Product Cost

Taguchi suggests that when the gain in SN ratio is sufficiently large, half of the gain should be allocated to reduce cost. Consider, for example, an automotive rotor/brake pad subsystem. Optimization motivated by need to reduce audible noise yielded a 20% improvement in performance in addition to eliminating the audible noise problem. The 20% performance improvement was captured as a 20% weight reduction that translated into about a 12% cost reduction.

Loss to Society

Minimize loss to society using tolerance design to balance internal company cost and external field cost. External field cost may be warranty cost or customer cost. For example, starting with current internal cost, balance company and customer costs by conducting tolerance design using the quality loss function. The methodology is too extensive to present here. Tolerance design is developed elsewhere in this handbook.

Taguchi suggests starting with the lowest-cost materials and parts as reasonable and upgrade only as necessary to minimize loss to society. This approach counters the engineer's tendency to overspecify materials and parts because it is "safe," albeit, potentially expensive. Again, tolerance design provides the means to determine what upgrades are needed.

44.6. Taming the Intersection of Six Sigma and Robust Engineering

Six sigma was born in Motorola as a structured methodology for improving existing manufacturing processes. Over the years, application of six sigma has been extended successfully to business and service processes. Mikel Harry is broadly recognized as the primary force behind popularizing six sigma [10]. He originally defined eight phases: recognize, define, measure, analyze, improve, control, standardize, and integrate with measure, analyze, improve, and control (MAIC) as the core.

Five phases have become the most frequently used structure for six sigma:

1. *Define.* Define project scope and plan.
2. *Measure.* Conduct measurement system evaluation, measure defect levels, understand customer needs.
3. *Analyze.* Dig, dig, dig in search of "root causes."
4. *Improve.* Eliminate root causes.
5. *Control.* Hold the gains through process control.

For more detail about six sigma in an easy-to-read, entertaining style, see *The Power of Six Sigma* [11]. The six sigma DMAIC process is usefully represented as a block arrow diagram to indicate the temporal sequence of activities.

DMAIC, IDDOV, and NPD block arrow diagrams represent temporal sequences of activities. However, the logic of the DMAIC process is fundamentally different. DMAIC consists of a sequence of activities for digging vertically down into the details of some portion of an existing product or process to find and eliminate the causes.

DMAIC and IDDOV intersect NPD differently. An unconventional representation of the relationship between DMAIC and NPD is depicted in Figure 44.11. DMAIC is oriented vertically to reflect its application for digging down into lower levels of an existing product to find and correct causes of problems. Define,

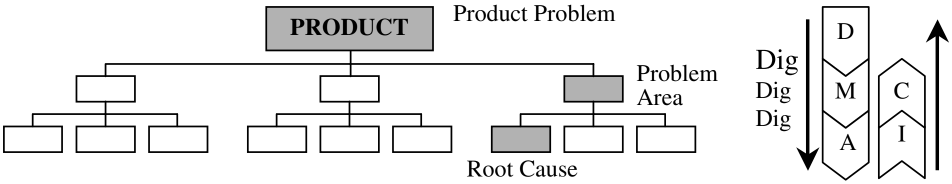


Figure 44.11
System architecture block diagram

measure, and analyze (DMA) flow vertically downward in search of root causes. *Improve* (I) flows vertically upward as improvements are made at lower levels and verified at higher system levels. *Control* (C) refers to the processes put in place to hold the gains.

The unconventional representation is intended to bring to mind logical rather than temporal relationships. DMAIC is used to find and fix problems in existing processes and products. Product applicability ranges from later portions of NPD when first-instance prototypes begin to exist and extend through the entire life cycle of customer use.

Complex products are typically represented as a hierarchy of system elements in two-dimensional block diagrams, as depicted in Figure 44.11. Define, measure, and analyze are applied to dig vertically down into increasing detail of a system to identify the causes of problems in existing processes. A hypothetical path from the product level to causes is indicated by the shaded system elements.

Once the cause is located, the methods within the improve phase are used to eliminate the cause. The process of verifying the improvement proceeds sequentially upward through the higher-level system elements to the product level. Then the methods within the control phase are utilized to hold the gains.

This unconventional representation of the relationship between DMAIC and NPD grew out of frustration from unsuccessful efforts to explain the differences between IDDOV and DMAIC to candidate DFSS black belts and to six sigma master black belts. The master black belts with the deepest knowledge of six sigma had the greatest difficulty breaking their “fixation” on the DMAIC paradigm. The accepted explanations that IDDOV is a creation process and DMAIC is an improvement process did not quell the continuing stream of questions that could extend for weeks. Analogies, different explanations, and alternative pictures did not seem to help. Finally, the vertical representation of DMAIC created “aha’s” of understanding at first sight.

DMAIC, founded on statistical processes used for controlling and improving manufacturing processes, has proved to be a very effective methodology for improving business, engineering, manufacturing, and service processes. Its effectiveness as a find-and-fix methodology for product problems is arguably less spectacular.

Finding and fixing the causes of product problems is a process of working on the symptoms of a fundamental weakness. Two types of disappointments often occur, identified here as type C and type WAM. Type C is the “chronic” recurrence of the same problem in program after program. Type WAM is the previously identified “whack-a-mole” engineering process of whacking down one problem only to have another problem pop up (e.g., fix audible noise but shorten life). Both types of problems have a common origin: lack of robustness.

Lasting improvement is achieved by focusing on the intended function, not by hiding the unpleasant symptoms of dysfunction. Improving the function is achieved by increasing the signal-to-noise ratio of the intended function to increase its robustness against the sources of variation that arise under real-world usage conditions. Robust optimization provides the methods for whacking all moles with one whack.

More generally, all of the tools and methods within IDDOV are useful in the DMAIC process when applied to product improvement. The following case study illustrates the use of a portion of the IDDOV tool set in a DMAIC-like find-and-fix process. The case study involves a type C chronic, recurring problem. Because the source of the problem was found to be in the software, the choice of control factors was not constrained by sunk investments in tooling. The freedom to select the appropriate control factors facilitated a complete resolution of the problem.

Case Study: Robust DMAIC

The case study entitled "Optimization of a Diesel Engine Software Control Strategy" by Larry R. Smith and Madhav S. Phadke [12] was presented at the American Supplier Institute Consulting Group, 19th Annual Taguchi Symposium, held in 2002. Genichi Taguchi selected the paper for the Taguchi Award as the symposium's best case study.

The case study addresses a long-standing, intermittent problem in diesel engine power train control system called *hitching* under cruise control and *ringing* in idle mode. Hitching and ringing refer to oscillations in engine rpm. The intermittent character and relatively rare occurrence of the problem had confounded engineers in numerous previous attempts to solve the problem.

The essence of the six sigma style case study was the unusual use of engineering methodologies in a DMAIC-like find-and-fix process to eliminate completely a long-standing and expensive problem that many previous efforts failed to fix. TRIZ was used to analyze the problem and identify software as the location of the cause. Robust optimization was used to improve performance by eliminating the problem permanently. Elimination of the hitching/ringing problem saved $8 million in warranty cost.

A fascinating element in the case study that is not emphasized in the paper is the method used to develop a noise strategy. Several engineers had the patience to drive several trucks for extended periods under a variety of conditions. Different driving profiles constitute important noise factors because they cause major changes to the load on the engine. The noise factors were selected from driving conditions that the team of drivers eventually found that would initiate hitching. Six noise factors were identified that related to acceleration and deceleration in 1-mph increments for different speed bands and rolling hills at different speeds.

The case study illustrates the potential power of an engineering-oriented DMAIC process for improving existing products. The notion is straightforward. The DMAIC process works best when engineering methods are used to improve existing engineered products and statistical methods are used to improve existing statistical processes. The heart of the new robust six sigma process would be robust engineering just as it is in IDDOV. The new process would have the power to change find and fix, fix, fix a chronic, reoccurring problem into find and eradicate the problem and prevent its reoccurrence, provided that the selection of control factors was not overly constrained.

In the conclusions section, the authors state:

> Many members of this team had been working on this problem for quite some time. They believed it to be a very difficult problem that most likely would never be solved. The results of this study surprised some team members and made them believers in the Robust Design approach. In the words of one of the team members, "When we ran that confirmation experiment and there was no hitching, my jaw just dropped. I couldn't believe it. I thought for sure this would not work. But now I am telling all my friends about it and I intend to use this approach again in future situations."

44.7. Conclusions

The dramatic benefits of robust optimization and IDDOV are characterized through quantification of improvements in terms of familiar metrics such as sigma level, product failure rate, and financial projections assuming the average gain in an SN ratio of 6 dB.

In addition, four new developments are introduced:

1. The IDDOV formulation of design for six sigma is shown to be a powerful implementation of Taguchi's quality engineering.
2. A new NPD process is described that takes advantage of IDDOV and two-step optimization.
3. An explicit formula for projecting reduction in warranty cost due to optimization is presented.
4. Innovation is fostered by the ability to optimize concepts and technology sets to reduce risk prior to a GOOD START commitment in NPD.

Finally, the tantalizing notion of creating a new robust DMAIC process focused on improving existing engineered products is introduced. Although this chapter focused only on product applications, nearly all of the notions and methodologies discussed are equally applicable to business, engineering, manufacturing, and service processes.

ACKNOWLEDGMENTS
The incredibly talented members of the ASI CG family made this chapter possible. I am especially indebted to my friends and colleagues, Subir Chowdhury, Shin Taguchi, Alan Wu, and Jim Wilkins for support, suggestions, and corrections. We are all indebted to Genichi Taguchi for his continuing stream of many outstanding contributions focused on helping corporations and engineers reduce the loss to society. I am also indebted to my good friend, Larry R. Smith, for continuing support over many years and for reading this manuscript and making valuable suggestions.

References

1. Subir Chowdhury, 2002. *Design for Six Sigma: The Revolutionary Process for Achieving Extraordinary Profits.* Chicago: Dearborn.
2. Genichi Taguchi, Subir Chowdhury, and Shin Taguchi, 2000. *Robust Engineering: Learn How to Boost Quality While Reducing Cost and Time to Market.* New York: McGraw-Hill.
3. Yuin Wu and Alan Wu, 2000. *Taguchi Methods for Robust Design.* New York: ASME Press.
4. Stuart Pugh, 1990. *Total Design: Integrated Methods for Successful Product Engineering.* Harlow, Essex, England: Addison-Wesley.
5. Genichi Taguchi, 1993. *Taguchi on Robust Technology Development: Bringing Quality Engineering Upstream.* New York: ASME Press.
6. Robert G. Cooper, 1993. *Winning at New Product Development: Accelerating the Process from Idea to Launch.* Reading, MA: Addison-Wesley.

7. How to bring out better products faster. Industrial Management & Technology issue of *Fortune*, November 23, 1998.
8. Joseph M. Juran and A. Blanton Godfrey, 1998. *Juran's Quality Handbook*, 5th ed. New York: McGraw-Hill p. 2.7.
9. Lloyd W. Condra, 1993. *Reliability Improvement with Design of Experiments*. New York: Marcel Dekker.
10. Mikel Harry and Richard Schroeder, 2000. *Six Sigma: The Breakthrough Management Strategy Revolutionizing the World's Top Corporations*. New York: Doubleday.
11. Subir Chowdhury, 2001. *The Power of Six Sigma: An Inspiring Tale of How Six Sigma Is Transforming the Way We Work*. Chicago: Dearborn.
12. Larry R. Smith and Madhav S. Phadke, 2002. Optimization of a diesel engine software control strategy. *Proceedings of the 19th Annual Taguchi Symposium,* sponsored by the American Supplier Institute Consulting Group.

This chapter is contributed by Barry Bebb.